3-2002

Integrated MANET Mutual Authentication System

Jason T. Ballah

Integrated MANET Mutual Authentication System

THESIS

Jason T. Ballah
First Lieutenant, USAF

AFIT/GCS/ENG/02M-01

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

Report Documentation Page

Report Date	Report Type	Dates Covered (from... to)
1 Mar 02	Final	Jan 01 - Mar 02

Title and Subtitle Integrated Manet Mutual Authentication System (IMMAS)	**Contract Number**
	Grant Number
	Program Element Number
Author(s) 1st Lt Jason T. Ballah, USAF	**Project Number**
	Task Number
	Work Unit Number
Performing Organization Name(s) and Address(es) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 P Street, Bldg 640 WPAFB, OH 45433-7765	**Performing Organization Report Number** AFIT/GCS/ENG/02M-01
Sponsoring/Monitoring Agency Name(s) and Address(es) AFIWC/IO (ACC) Attn: Ms Carol Hiltpold 102 Hall Blvd Ste 350 San Antonio TX 78243-7039	**Sponsor/Monitor's Acronym(s)**
	Sponsor/Monitor's Report Number(s)

Distribution/Availability Statement
Approved for public release, distribution unlimited

Supplementary Notes

Abstract
The Integrated MANET Mutual Authentication System (IMMAS) provides implied mutual authentication of all routing and data traffic within a Mobile Ad Hoc Network (MANET) by combining Elliptic Curve Cryptography, a public-key cryptosystem, with the Dynamic Source Routing (DSR) Protocol. IMMAS provides security by effectively hiding network topology from adversaries while reducing the potential for, among other things, traffic analysis and data tampering, all while providing a graceful degradation for each of the authentication components. Current research in MANETs tends to focus primarily on routing issues leaving topics such as security and authentication for future research. IMMAS focuses on achieving a higher level of security with the potential for substantial increases in efficiency of processing power and bandwidth compared to alternative exterior authentication mechanisms tacked on after protocol development and standardization.

Subject Terms
MANET, Mutual Authentication, Elliptic Curve Cryptography, Dynamic Source Routing, Network Security

Report Classification unclassified	**Classification of this page** unclassified
Classification of Abstract unclassified	**Limitation of Abstract** UU
Number of Pages 115	

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense or the United States Government.

AFIT/GCS/ENG/02M-01

INTEGRATED MANET MUTUAL AUTHENTICATION SYSTEM (IMMAS)

THESIS

Presented to the Faculty of the Graduate School of Engineering and Management

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Jason T. Ballah, B.S.

First Lieutenant, USAF

March, 2002

AFIT/GCS/ENG/02M-01

INTEGRATED MANET MUTUAL AUTHENTICATION SYSTEM (IMMAS)

Jason T. Ballah, B.S.

First Lieutenant, USAF

Approved:

Rusty O. Baldwin	1 Mar 02
Major Rusty O. Baldwin Thesis Advisor	Date
Richard A Raines	1 Mar 02
Major Richard A. Raines Committee Member	Date
Gregg H Gunsch	1 MAR 02
Dr. Gregg H. Gunsch Committee Member	Date

Acknowledgements

I would like to thank my thesis advisor, Major Baldwin, whose support and direction played a major role in successfully bringing this thesis research to fruition. I would also like to recognize my thesis committee members, Major Raines and Doctor Gunsch, for their assistance and suggestions throughout this thesis process. The help of these instructors was greatly appreciated.

Jason T. Ballah

Table of Contents

Page

List of Figures

List of Tables

Abstract

The Integrated MANET Mutual Authentication System (IMMAS) provides implied mutual authentication of all routing and data traffic within a Mobile Ad Hoc Network (MANET) by combining Elliptic Curve Cryptography, a public-key cryptosystem, with the Dynamic Source Routing (DSR) Protocol. IMMAS provides security by effectively hiding network topology from adversaries while reducing the potential for, among other things, traffic analysis and data tampering, all while providing a graceful degradation for each of the authentication components. Current research in MANETs tends to focus primarily on routing issues leaving topics such as security and authentication for future research. IMMAS focuses on achieving a higher level of security with the potential for substantial increases in efficiency of processing power and bandwidth compared to alternative exterior authentication mechanisms tacked on after protocol development and standardization.

INTEGRATED MANET MUTUAL AUTHENTICATION SYSTEM (IMMAS)

I. Introduction

Due to recent performance advancements in computer and wireless communications technologies, mobile wireless computing is becoming increasingly widespread. One type of wireless network that is quickly evolving is the Mobile Ad Hoc Network (MANET). Unlike other mobile network paradigms, such as cell phone networks with fixed radio towers and centrally accessible routers and servers, MANETs have dynamic, rapidly-changing, random, multi-hop topologies composed of bandwidth-constrained wireless links and no centrally accessed routers or servers. This networking paradigm reflects a level of mobility as yet unrealized in the world of networks. A MANET seeks to take computing resources ranging from pocket-sized wireless Personal Digital Assistants (PDA) to full-size wireless-network capable computers and expand the capabilities to the level of service provided on current wired Local Area Networks (LAN). This will be performed even as these computers may be travelling in vehicles or aircraft with little or no fixed network routers or infrastructure available.

1.1 Background

Ongoing studies of MANETs pose many intriguing and challenging research areas, but one important aspect that tends to be ignored is network security. Since MANETs are made up entirely of wireless mobile nodes, they are inherently more susceptible to security threats compared to fixed networks [ZaH99]. Access to wireless links is virtually

impossible to control thus adverse security events such as eavesdropping, spoofing, and denial of service attacks are more easily accomplished. In order for prospective MANET users to even consider this new paradigm in networking as a substitute for providing their mobile networking services, these security risks must be reduced to an acceptable level while maintaining an acceptable Quality of Service (QoS) and network performance.

Security problems are compounded by the fact that MANET nodes are working with a restricted amount of bandwidth and are typically limited in computing and battery resources. This places a practical limit on the security policies and procedures that can be implemented versus what are available for fixed networks. Thus, it is imperative to have efficient security mechanisms.

Suppose an Air Expeditionary Force deployment requires a bare base setup and an Expeditionary Combat Support network to support thousands of military personnel. As the base is being setup, a MANET would greatly reduce the need for wire to be laid to every office space. It is obvious that authentication of each and every MANET node is of utmost importance since this base is in or near enemy territory. The probability of an adversary posing as a valid MANET node to gain access to the network cannot be ignored. Furthermore, as the base grows and additional MANET-capable computers are added to the network, the limited bandwidth can quickly become congested. Thus, an authentication mechanism must ensure the level of security needed for a particular situation is obtained, however it must not take up too much of the limited MANET bandwidth.

The previous example shows why the problem of inter-node authentication of data and routing information is a pressing issue to be solved. Several solutions for this problem

have been designed for fixed and wireless Local Area Networks (LAN) [BaB98, KaN93, MTH92, NaS78, PaS98, SaA99, DaA99], but no standard solution has been developed specifically for MANETs. Currently, only a handful of MANET researchers have even addressed the general problem of MANET security [ZaH99, YNK01, VaA00, HBC01, JaC99] and of those listed, only one proposed solution to this authentication problem has been fully developed and published - the Internet MANET Encapsulation Protocol (IMEP). IMEP provides authentication for a MANET's routing data through the exchange of additional IMEP packets which are transmitted with every routing packet [JaC99, ICP00].

1.2 Goals

The overall goal of this research is to develop an effective authentication mechanism for a MANET that can be incorporated directly into the MANET routing protocol. This mechanism will achieve a given level of authentication and security while not eroding the QoS. The hypothesis is that an efficient authentication mechanism providing a high degree of security, versatility, and adaptability is one that is directly integrated into the MANET routing protocol versus performing bulk encryption at the time of transmission. The analysis will encompass fixed network authentication mechanisms, current encryption technologies, and a new routing protocol to be developed for this research. In order to attain the above stated goal, the following objectives will need to be met:

1. Develop and integrate a mutual authentication system into an existing MANET routing protocol, and
2. Determine what performance impact the authentication system has on the MANET.

Authentication and security for MANETs is an area of research all by itself, thus many researchers have dedicated their efforts to developing protocols that will enable the MANET paradigm to work properly in an unsecured environment. This implicitly assumes the data is authenticated and secure. This study will provide a security and authentication system as well as determining the impact of such a system on network performance.

1.3 Document Overview

This chapter provides a basic introduction and background to the problem of authentication within Mobile Ad Hoc Networks. It defines the goals of this research. Chapter II provides background information in the areas of network authentication, MANET routing protocols, as well as authentication mechanism integration into routing protocols. Chapter III contains the methodology this research used to approach the problem. Chapter IV describes the design and implementation of the authentication system developed for this research. Chapter V presents the results obtained from the experiments and provides an analysis of those results. Chapter VI describes research conclusions as well as areas for further study.

II. Literature Review

2.1 Introduction

This chapter examines the current Mobile Ad Hoc Network (MANET) paradigm as described by the Internet Engineering Task Force (IETF) MANET working group [CaM99]. In particular, the Dynamic Source Routing (DSR) protocol is discussed along with a review of previous research implementations of DSR. Authentication mechanisms developed for wired and wireless networks are explored to provide an overview of what is currently available. The role encryption mechanisms play in various authentication systems is also covered. Lastly, current authentication and security research and mechanisms developed for the MANET paradigm are presented.

2.2 MANET Routing Protocols

There are currently between 10 and 15 MANET routing protocols recognized by the IETF MANET working group. However, in 2001, the IETF MANET working group accepted two MANET Routing Protocols as experimental standards. These protocols are the Ad hoc On-demand Distance Vector (AODV) routing protocol and the Dynamic Source Routing (DSR) protocol [JMH01a, PRD01]. Either protocol would work equally well for this research since they are only providing the baseline by which this research will be measured and both have strengths and weaknesses. So, the criteria used to select one of these two protocols was based on an expert opinion as to which would provide a desirable military scenario as well as what results from this research would arguably prove to be the

most important to the MANET community. DSR thus became the protocol of choice for this research.

DSR was designed to take advantage of the multitude of information traversing a MANET by allowing all nodes to cache multiple routes to any particular destination node by promiscuously "snooping" and "tapping" information about the routing layout of the network from packets traversing the network. DSR also was designed to be able to take advantage of Medium Access Control (MAC) protocols which allow uni-directional communication for the routing of packets much like the idea of transmitting User Datagram Protocol (UDP) packets or multi-media streaming transmissions. However, this capability is unnecessary when DSR is implemented over MAC protocols such as MACA [Dar90], MACAW [BDS94], or IEEE 802.11 [IEE00], which require bi-directional communication between nodes for the exchange of RTS and CTS packets prior to the exchange of a data packet. On the other hand, AODV uses a table-driven routing scheme, which requires all reply packets to follow the reverse route path and only maintains one route to a particular destination. This characteristic of AODV leads to a larger number of route discoveries when links for a route become unusable, especially in a lightly loaded system [DPR01]. Due to these differences, the DSR protocol was chosen for this research as it tends to fit the highly unpredictable military ad hoc environment better than AODV.

2.3 Dynamic Source Routing (DSR) Protocol

The DSR Protocol was designed by Johnson, Maltz, Hu and Jetecheva specifically for use in multi-hop wireless ad hoc networks with mobile nodes [JMH01a]. This research will look at the March 2001 version of the DSR specification [JMH01a]; however, it should

be noted that a new version was released in December 2001 [JMH01b]. The differences in this new version have no bearing on the outcome of this research. Although simple enough to be a link-layer routing protocol, DSR is specified to be implemented at the network layer in the ISO 7-layer model as seen in Figure 2.1. This is due in large part to the requirement of supporting multiple network interfaces of different types forming an ad hoc network [JMH01a]. In a wired network, the protocols used at the different layers of the ISO model tend to promote horizontal communication between layers to increase conservation of router resources, whereas in MANET protocols such as DSR vertical communication between the layers is promoted to increase the conservation of bandwidth [CMC99]. In other words, MANET routing protocols tend to gather, share, and consolidate information in the packet header from between the layers as much as possible versus each layer having its own specialized set of information in its own header that is unusable by any other layer. More specifically, a MANET routing protocol will often use information from the TCP/UDP header, IP headers, as well as knowledge of the MAC layer to make a more efficient routing header.

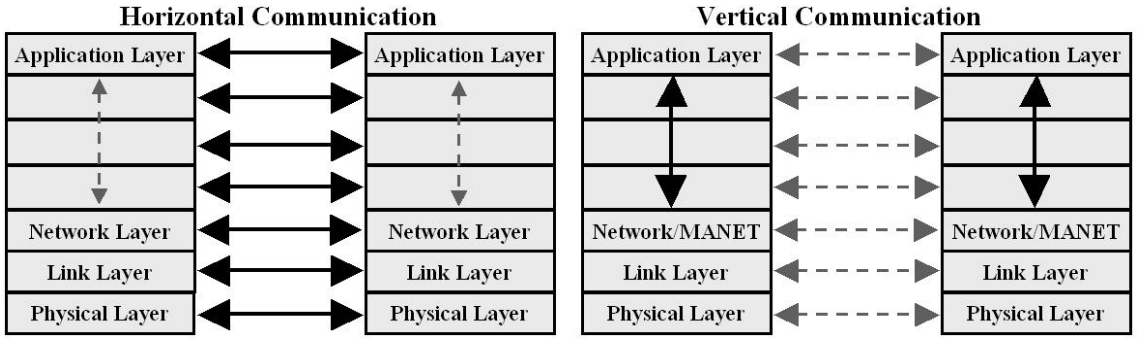

Figure 2.1. Network Communication Patterns

The DSR protocol is based on the precept that each data packet will follow a source route that is discovered and maintained by a source node and stored in the header of all data

packets to be sent from the source to a destination. Once a source node has discovered a route, a packet is sent to each node along the route's path to the destination node using the links defined by the source route. Although IEEE 802.11 requires bi-directional communication for its RTS/CTS hand-shaking protocol between two nodes, DSR can also be used in MAC-layer protocols that do not use bi-directional communication. A benefit of this on-demand routing protocol is that routing information traversing the network approaches zero in an approximately stationary ad hoc network, and in a highly mobile ad hoc network will only get as large as the number of routes being used. The DSR protocol uses two basic on-demand mechanisms, Route Discovery and Route Maintenance, to discover and maintain routes within a MANET [JMH01a].

2.3.1 Route Discovery. Route Discovery is used by a source node to query all directly and indirectly linked nodes about a route to a specific destination node. A route is found by the source node sending a route request packet containing the address of the source and destination nodes. Each intermediate node checks to see if a route to the destination node is in its own route cache. If so, the intermediate node attaches its address and the list of addresses for the route to the destination node into a route reply packet which is sent back to the destination node. If the intermediate node's route cache does not have a route to the destination node, its address is appended to the list of addresses in the route request packet and the packet is forwarded on to all other nodes within its transmission range. All nodes receiving this transmission will process the route request packet. If the node has already received the route request packet then all subsequent receptions of this request are discarded. Using this scheme, the first positive response

back to the source node is ideally the preferable route because it implies that this route is either the shortest or the least congested route. This could be a route discovered from an intermediate node's route cache along the way or a reply from the targeted destination node.

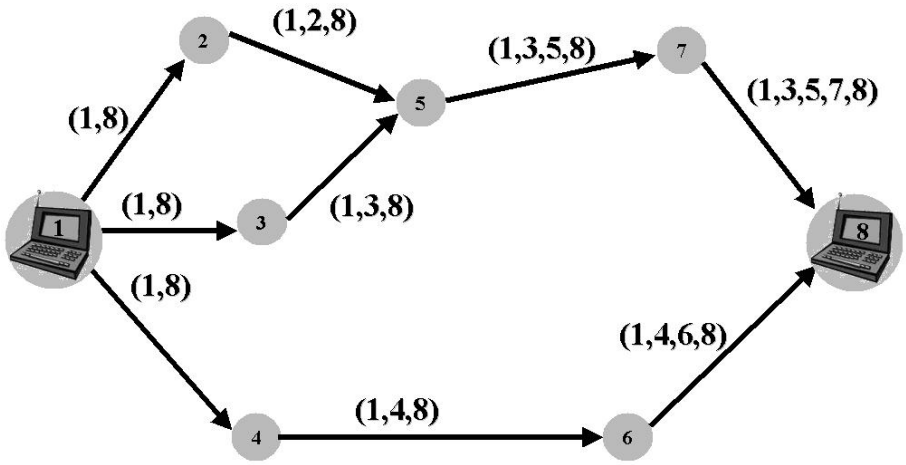

Figure 2.2. DSR Route Discovery Request

To demonstrate this process, consider Figure 2.2 where node 1 has a packet to transmit to node 8 and no nodes have a route to node 8 in their route cache. A request packet is sent from node 1 and received by nodes 2, 3, and 4. Since they do not have a route in their cache then they append their address to the request packet and retransmit the request. Since nodes 2, 3, and 4 have all just processed this particular request, they will discard the re-transmitted requests from each other. Figure 2.2 depicts node 5 receiving the transmission from nodes 2 and 3; however, node 5 will ignore the transmission from node 2 since it received the transmission from node 3 previously with the same request. Node 6 will receive the transmitted request from node 4 and re-transmit since a route to 8 is not in the route cache. Node 7 will receive the request transmission from node 5 and after adding its address to the request packet re-transmit it as well. Node 8 will receive the

request transmission from node 6 and discard any future receptions with the same request such as will be received from node 7. Node 8 will send a reply packet back to node 1. If a MAC protocol, such as IEEE 802.11, requires bi-directional communication between two nodes for the exchange of hello/request and acknowledgment packets, then the reply packet will follow the reverse path. In the case of Figure 2.2, the reply can be sent back to node 1 with the path from node 8 to 6 to 4 to 1 as shown in Figure 2.3 since this is quicker and more reliable than determining another route. Otherwise, it is possible that DSR can send a reply packet back to the source node using a different route than by which the request packet traversed to the replying node.

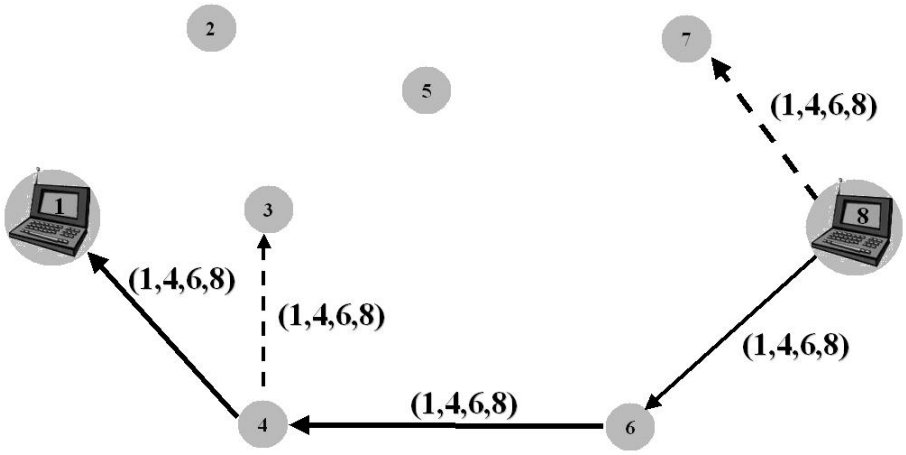

Figure 2.3. DSR Route Discovery Reply

This use of uni-directional links is important because the highly dynamic and diverse military environment many times produces scenarios where the transmission power or range from node 1 to node 2 is not the same as from node 2 to node 1. Should a standard MAC protocol for wireless MANETs be developed to take advantage of this uni-directional idea, DSR will be capable of using it. Another outcome from the scenario depicted in Figure 2.2 and Figure 2.3 and is that nodes 3 and 4 now have routes in their cache from themselves to

node 8. Therefore, both nodes will not forward the request, but instead will wait a short amount of time (a function of the size of the final source route). In this way, node 4 will respond first with a reply packet to node 1 specifying the route 1 to 4 to 6 to 8. Node 3 will overhear node 4's reply transmission and will cancel its own reply. The reply to node 1's request is much shorter and not all nodes needed to handle a request to re-transmit thus lowering network contention.

Once a route to a destination node has been discovered and a route reply packet is sent back to the source node, the route information is placed in the header of all data packets to be sent to the destination node. This information is also placed in the source node's route cache for later use. If a node is in promiscuous mode, route information can be gathered from transmitted request packets, reply packets, or source route data packets, and added to that node's route cache. This allows nodes to maintain the status of the highly dynamic network through both active and passive means. The disadvantage is that this mechanism is optional in the specification. This allows various implementations of the route cache to contain multiple route caching strategies that can have a large effect on the performance of the protocol [MBJ99]. Taking into account the size of the route, how the route was discovered, and when it was discovered all contribute to how efficient the cache is at maintaining valid routes. A less efficient cache will contain more invalid routes thus increasing the overall network load through route discoveries and route maintenance versus efficiently using promiscuous information.

2.3.2 Route Maintenance. Route Maintenance is invoked by a source node if a route is found to have a broken link during the transmission of packets to a destination.

This maintenance mechanism allows the source node to either find a new path within its route cache or invoke the Route Discovery mechanism to find a new route for the lost packets (and those still waiting to be transmitted). If a route is not found, the source node will continue to use the Route Discovery mechanism and send out route request packets using an exponential back-off algorithm to determine the amount of time to wait before sending the next Route Discovery. If the exponential back-off algorithm reaches a maximum time limit, the protocol will return a failure message to the upper layer.

2.3.3 Conceptual Data Structures. There are four conceptual data structures included in the draft specification of DSR [JMH01a]. These include the Route Cache, Route Request Table, Send Buffer, and the Retransmission Buffer. Each of these data structures are briefly described below.

1. Route Cache. The route cache data structure is used to store routes to destination nodes. The DSR specification allows a lot of leeway in the implementation of this data structure. It permits one or more routes to any destination with a fixed or variable sized cache. The implementation of this data structure is left up to the designer as is the algorithm used to determine what route to use when searching the cache for a destination. Route information can be gathered from request packets, reply packets, and/or the source route information in data packets.

   The route cache is the "bread and butter" of the DSR protocol. The high delivery ratios and low routing overhead seen in previous research [BMJ98, DPR01, MBJ99], can be attributed largely to the optimized route-caching implementation. This is indicated by the performance difference of DSR and AODV. AODV's basic functions

are similiar to DSR, yet the AODV table-caching strategy of a single link to a destination has a much larger routing overhead [DPR01]. Different implementations of the DSR route cache create a large variation of results for the basic performance of the protocol. This will be explained in more detail in Chapter IV along with the implementation decisions used for this research.

2. Route Request Table. The Route Request Table maintains information about all route requests originating from that node. The information maintained includes the time the last outstanding route request was initiated for a target node, the number of consecutive route requests for that target node, the wait time (determined by the exponential back-off algorithm discussed in Section 2.2.1.2) before the next request can be sent for a target node, and time to live information used in the last request for a particular target node. This information is used along with the exponential back-off algorithm for sending future route requests should a route reply not be received from an initial route request.

3. Send Buffer. The Send Buffer is used to hold data packets for a particular destination in a First-In-First-Out (FIFO) queue structure until the node has a source route to the destination. Each node has a separate send buffer for each destination being used. Packets are held in this buffer for a certain amount of time and are dropped if a route is not found.

4. Retransmission Buffer. The Retransmission Buffer holds a copy of transmitted data packets until an acknowledgement is received from the next node in the route. This permits the retransmission of data packets should an error or collision occur.

2.3.4 Previous DSR Research Implementations. DSR was designed and implemented at Carnegie Mellon University (CMU) as part of the Monarch Project using the freely available Network Simulator-2 (NS-2) [BMJ98, MBJ99]. It was compared against AODV as described in [DPR01] using a slightly modified version of CMU's NS-2 DSR model. The DSR protocol was implemented in OPNET by the National Institute for Standards and Technology (NIST). However, there are no performance results published comparing that implementation to the other implementations discussed here. The next two sections will discuss some of the more critical implementation decisions used for the above research implementations.

1. Node Movement Patterns. There are a number of node movement patterns referenced throughout literature. Two of these include the random waypoint model [BMJ98] and the random direction model [RSM01]. The random waypoint model was the movement model of choice for all of the DSR NS-2 implementations and is said to produce the most random movement of nodes [BMJ98].

   (a) Random Waypoint Model. The random waypoint model works by initially distributing all nodes uniformly within the simulation area. A random waypoint is then chosen for a node to move to. Nodes wait a PAUSE time before moving to the chosen waypoint. Once the PAUSE time has elapsed the node will randomly choose a speed between 0 and MAXSPEED and proceed at that rate to the chosen waypoint. Once the node has reached the waypoint another waypoint is chosen and the process is repeated. More information covering the implementation and problems of this model can be found in Appendix A.

(b) Random Direction Model. The random direction model is similar to the random waypoint model except that instead of choosing a waypoint somewhere in the simulation area, the model first chooses a direction to travel, then chooses a point within the simulation area that is in that direction. This behavior helps alleviate the problem of converging to the center as seen in the random waypoint model and described in [RSM01].

2. Critical Simulation Parameters. Other critical simulation parameters used by previous research include simulation area, number of nodes, number of source nodes, node speed, node pause times, packet interarrival times, payload sizes, transmission range, and data rates.

(a) Simulation Area. Simulation areas varied from 1000 x 1000 meters, 1500 x 300 meters, 600 x 300 meters, to 670 x 670 meters. Simulation areas such as 1500 x 300 meters simulates a highway type of structure with the cars as the MANET nodes. This type of simulation tends to force more linear type of routes with longer path lengths. On the other hand, the 670 x 670 meter area simulates a city or office type of structure where nodes can move freely around each other. This type of simulation tends to have fewer bottlenecks and network congestion due to spatial diversity as well as providing shorter route lengths. The 600 x 300 meter simulation area provides a little of both of the aforementioned simulation areas characteristics.

(b) Nodes. The number of nodes used in previous implementations included 14, 20, 50, and 100 nodes. Published data for DSR implementations, however, primarily came from 50 nodes being placed in one of the simulation areas described above.

(c) Source Nodes. The number of source nodes was varied to produce different workloads on the network. The typical implementation used 20 of the 50 nodes as source generators, but 10, 30, and 40 source nodes were also used to portray how the DSR protocol works under various loads. One implementation even used 14 originating nodes, but six of the nodes used two source generators to produce a total of 20 sources. In each of these cases, peer-to-peer connections were used. In a peer-to-peer scenario, a destination node is determined at the beginning of the simulation and data packets from that source are sent to this destination throughout the simulation.

(d) Node Speed. Node speed was another parameter that was varied depending on the research goals of the implementation. The maximum speed that was typically used was 20 meters/second and the speed was randomly chosen between 0 and that maximum speed. However, some of the research did use 1 and 5 meters/second to portray different effects on the network at slower average speeds.

(e) Pause Time. Node pause time was also used by the node movement model to determine how long a node would wait prior to starting movement to a particular destination. Pause times were varied for all of the implementations to include 0, 30, 60, 120, 300, 600, and 900 seconds. A pause time of zero means that the node is in constant movement and since all of the research simulations were

run for 900 seconds the pause time of 900 seconds means that the nodes were stationary.

(f) Packet Interarrival. The research implementations of DSR primarily used packet interarrival times of 0.25 seconds. Some research increased this to 0.5 and 1.0 seconds to decrease simulation time, but published results all used 4 packets per second.

(g) Packet Sizes. Packet sizes varied from 64 bytes, 512 bytes, to 1024 bytes. The implementation with 1024 byte packets [BMJ98] found that this was too large for the simulation area and caused too much congestion. Thus, published data for that implementation used 64 byte packets. Other implementations used 512 byte packets. However, as pointed out by Jeff Boleng [Bol00], over 60 percent of data packets on the internet are 44 bytes or less (before 20-byte IP headers). Thus, 64-byte packets seem to be the more "realistic" simulation scenario since the MANET is attempting to re-create the Internet in a mobile, wireless environment.

(h) Transmission Range. All of the simulation implementations of DSR used a nominal transmission range of 250 meters. However, this is an optimal range and does not take into account realistic terrain and environmental factors that would be seen with an actual implementation. This is an issue that was discussed at the MOBIHOC 2001 conference in Long Beach CA on 4-5 October, 2001 in various technical sessions. It was suggested that a more realistic approach is to simulate a transmission range of 100 meters in the same simulation area. This would account for sub-optimal conditions as well as stressing the protocols more

by forcing 4-5 average hops versus 2-3 depending on the size of the simulation area. While this is an interesting idea, all of the literature reviewed for this research used a transmission range of 250 meters and this research followed that trend.

(i) Data Rate. The last critical simulation parameter to be discussed is the data rate. The data rate that has been overwhelmingly used by researchers thus far is 2 Mbps using IEEE 802.11. There were arguments presented at MOBIHOC 2001 to increase this rate to the latest IEEE 802.11b specification capabilities of 5.5 or 11 Mbps. It was, however, argued that since the higher frequency of 5 GHz is not available in all areas of the world that research should maintain a standard data rate that everyone can use.

2.4 Authentication Mechanisms

As network technology has grown and matured, the need for network security and authentication has grown as well. Many authentication systems have been developed for both wired and wireless networks. One thing they all have in common is a central authentication server known to be trusted as well as operational. A MANET does not have the luxury of a central server to rely on. This greatly limits the potential use of these types of authentication schemes. However, a discussion of these schemes provides a background on current authentication systems and points out areas that can be used for a MANET authentication system. These systems include Kerberos, KryptoKnight, various wireless Asynchronous Transfer Mode (ATM) authentication mechanisms, as well as the Mutual Authentication, Confidentiality, and Key Management System.

2.4.1 Kerberos / KryptoKnight. The Kerberos system was developed at MIT in the late 1980's for project Athena [KaN93]. It has matured over the years and has become a mainstay for many wired authentication systems. Kerberos has also recently been ported to a wireless paradigm for Local Area Networks [PMK00]. The Kerberos model is based in part on Needham and Schroeder's trusted third-party authentication protocol [NaS78]. In this type of system, the authentication server is a trusted third party between a client and a network service provider. Kerberos uses one-time session keys sent from the authentication server to the client and the verifier to provide mutual authentication. The network must be able to prove that the person using a ticket is the same person to whom the ticket was issued. Kerberos performs this authentication in the manner depicted in Figure 2.4, which was adapted from [KaN93].

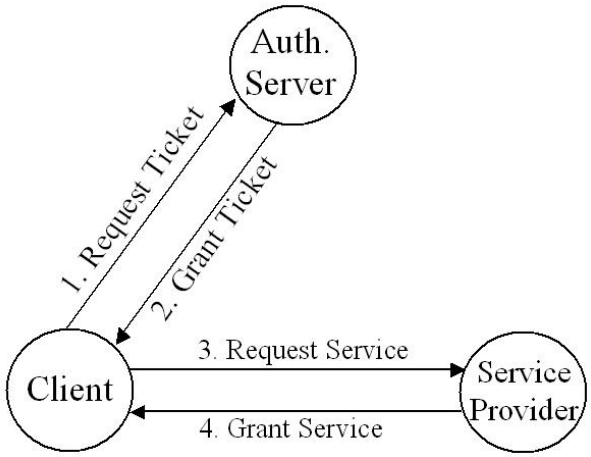

Figure 2.4. Kerberos Authentication [KaN93]

1. The client requests a ticket from an authentication server.
2. The authentication server produces a one-time session key and creates a ticket with it and some timestamp information. This ticket is sent back to the client.

3. The client produces an authenticator from the session key in the ticket and sends it along with a request to the desired service provider.

4. The service provider decrypts the ticket and uses the session key to decrypt the authenticator (which can only be properly decrypted by the desired service provider). This scheme allows the service provider to match the username and address of the ticket to the username and address of the client.

5. The service provider then sends a reply, encrypted with the session key, back to the client to confirm that the intended service provider received the request and is who the request was actually sent to. This successfully provides mutual authentication between the client and service provider.

The KryptoKnight security system was built using Kerberos as a stepping-stone and it also uses three network entities to perform authentication. However, instead of using the Needham and Schroeder scheme, it uses a family of novel authentication and key-distribution protocols to achieve the same mutual authentication between the network client and network service [MTH92, BGH95]. Figure 2.5 portrays the relationship between peer entities and KryptoKnight components involved in a program authentication. In this case, both the Initiator and the Responder will first authenticate with the KryptoKnight Authentication System (AS). Then, information from those authentications will be shared with the Initiator and the Responder in the form of keys which will be used by the two entities to communicate with each other [MTH92, BGH95].

Kerberos and KryptoKnight are both unsuitable for MANET applications due to the requirement for central servers to perform authentication and verification along with

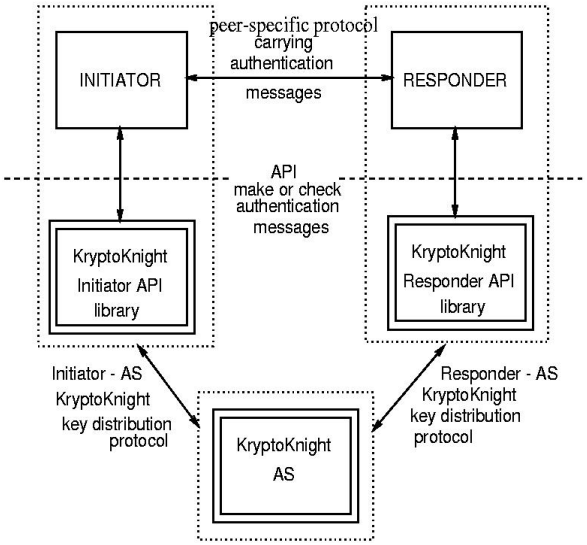

Figure 2.5. KryptoKnight Authentication [BGH95]

the fact that a MANET needs an authentication system that does not rely on any one particular node or server. However, the implementation of mutual authentication was shown to be an extremely important component of these authentication systems and it is something that should be integrated into any network authentication system. It should also be pointed out that Kerberos and KryptoKnight were designed to provide mutual authentication without building that authentication around encryption of anything more than the ticket, request, and reply. They are designed to use encryption as an add-on to provide the integrity and confidentiality needed to ensure the authentication is valid, but this is not the back-bone of these systems.

2.4.2 Authentication Protocols for Wireless ATM Networks. Patiyoot and Shephard reviewed a number of authentication protocols for wireless ATM networks including Challenge/Response, Secret Shared Information (SSI), Public/Secret Key, KryptoKnight, Random Key, Server Key, Security Agent, and Sequence Numbers [PaS98]. They con-

clude the protocol chosen depends on the network node's computational ability, memory resources, network bandwidth, real-time constraints, expenses, and security requirements. There were only three methods of authentication discussed that did not require the use of a third party. These were Challenge Response, SSI, and Public Key Cryptography as shown in Table 2.1. While they did not come to any conclusions on what protocols were best, the positive and negative comments they provided for each of the protocols gives further insight into what is available as well as the vulnerabilities of each of the protocols. For instance, they point out that although the KryptoKnight system's qualities allow it to be placed at any layer to accommodate security related information, the requirements for long-life power supplies and synchronized clocks with the KryptoKnight system make it infeasible for wireless systems.

Table 2.1. Methods of Authentication for Wireless ATM Networks [PaS98]

	Secret Key	Public Key	One-way Function	Third-Party Needed
Challenge/Response	Yes		Yes	No
SSI	Yes		Yes	No
Public/Secret Key	Yes	Yes	Yes	No
Kryptoknight		Yes	Yes	Yes
Random Key		Yes	Yes	Yes
Server Key		Yes	Yes	Yes
Security Agent		Yes	Yes	Yes
Sequence Number	Yes		Yes	Yes

2.4.3 Mutual Authentication, Confidentiality, and Key MANagement (MACKMAN).

The MACKMAN system takes an entirely different approach to authentication. While Patiyoot and Shephard discussed public key authentication to a limited extent with the ATM wireless authentication protocols, the MACKMAN system was designed around it. This system uses a fairly new form of public key cryptography called Elliptic Curve Cryp-

tography (ECC) [BaB98]. Other implementations of Public Key Cryptography such as Rivest-Shamir-Adleman (RSA) and Diffie-Hellman require the use of more processing and battery resources, as well as bandwidth [BSS99]. Any use of this type of cryptography in authentication would be prohibitive for network nodes with limited processing capabilities or network bandwidth. ECC, discussed in a later section, can be used for all data traffic in the MACKMAN system since it provides a comparable level of security while greatly reducing the processing and bandwidth requirements compared to conventional public key cryptography systems [BaB98].

2.4.4 Authentication Mechanisms employed by IEEE 802.11 and IPSEC. Before ending this discussion on authentication mechanisms, a brief look should be taken at what authentication mechanisms are available at lower ISO layers. The IETF working group for IP Security (IPSEC) has been developing security procedures and protocols that can be implemented by IPv4 and IPv6 [TDG98]. In addition, the IETF IPSEC Remote Access Working Group is working on a proposal for a Pre-IKE Credential Standard (PIC) which authenticates devices that are authorized to communicate with the system via Internet Key Encryption (IKE) and a system's secure IPsec gateway [Mil01]. However, IPSEC only provides one-way authentication of the sender. It does not provide mutual authentication between source and destination nodes or authentication for any of the intermediate hops between the source and destination. IEEE 802.11b also provides a limited shared-key authentication mechanism using the Wired Equivalent Privacy (WEP) encryption mechanism for the transmission of authentication frames at the link layer [IEE99]. This process provides mutual authentication between two neighboring nodes, but it does not provide

authentication between the original source node and the final destination node. The WEP option in IEEE 802.11b has also been criticized as being a rather weak algorithm that is fairly easy to break, so the security level for using this authentication protocol is considered low [Mil01].

2.4.5 Onion Routing. Onion Routing is a concept implemented by [GRS99] where the source node of a data packet will determine the route a packet will take, then one-by-one continue adding layers of encryption using the public keys of the next hop. Thus, as the data packet arrives at each location, a layer of encryption is removed and the packet is sent to the next hop. In this way the last layer of encryption will be removed at the destination node and the packet will be in plain-text. While this type of routing can not be used directly in this research due to the overhead, the idea of encryption so only the appropriate next hop can decrypt is used.

2.5 Encryption in Authentication

Good authentication allows entities to provide evidence that they know a particular secret without having to reveal that secret [PaS98]. While encryption is not required for an authentication mechanism, it is an extremely useful tool to provide the additional security of privacy, integrity, and non-repudiation that is needed to properly validate an authentication process. For Kerberos and KryptoKnight, encryption is a secondary add-on. Those protocols are built entirely on the authentication process with encryption as a possible option [KaN93, MTH92]. On the other hand, the MACKMAN system uses public key cryptography as its primary modus operandi. Due to security risks associated with wireless

networking as well as the possible military applications for MANETs, an authentication built around cryptography such as with MACKMAN is a much more useful approach. The next sections present the types of encryption typically used in authentication systems for wired and wireless networks.

2.5.1 Wired Networks. Kerberos and KryptoKnight authentication are able to use encryption algorithms such as the Data Encryption Standard (DES), Message Digest 5 (MD5), and Rivest-Shamir-Adleman (RSA) if needed for bulk encryption [KaN93, MTH92]. There are many other systems available as well that use encryption to implicitly gain authentication between two sources based on the knowledge of encrypted keys. The Secure Shell (SSH) [YKR01], Secure Socket Layer (SSL) [DaA99], Virtual Private Networks (VPN) [EWS98], and Virtual Local Area Networks (VLAN) [VLAN98] are all examples of protocols that use encryption to tunnel data traffic from point A to point B on a network or over the Internet and gain authentication through the use of cryptography. However, many of these protocols are only suited for wired networks since they can create a large amount of overhead and take up a large percentage of the available bandwidth as well as processing power.

2.5.2 Wireless Networks. The amount and type of encryption that can reasonably be performed in wireless networks is dependent on many factors including node size, processing power, bandwidth, and battery power. Most any form of encryption that can be used in a wired network environment can also be used in a wireless environment as long as the network factors mentioned above are adequate. However, this is typically not the case and many wireless networks, including MANETs, work under severe resource limita-

tions. The form of encryption that quickly rises to the top for consideration in wireless networks is Elliptic Curve Cryptography (ECC) as used in MACKMAN. This public key cryptosystem, as its name suggests, is based on elliptic curves and was introduced in 1985 by Neal Koblitz [Kob87]. Its security is based on the intractability of solving discrete logarithm problems [BSS99, RSA01, Men93]. This cryptosystem provides the convenience of privacy, integrity, and non-repudiation of data thus providing implicit authentication. Its implementation provides roughly 10 times faster processing with about one-third smaller data expansion compared to other forms of public key cryptography such as RSA [Cer97]. In fact, [BSS99] points out that ECC with a key strength of approximately 160 bits is equivalent to an RSA key strength of 1024 bits. By using this cryptosystem, mutual authentication across wireless nodes can be achieved for every data packet that is signed with the sender's private key and encrypted with the destination node's public key. In the past, this type of mutual authentication was only feasible for small amounts of data such as keys for symmetric cryptography which, in-turn, were used for larger amounts of data. This system of authentication could not be used for network or streaming communications due to the large increase in the size of overhead for the data being sent. With ECC integrated into the MANET routing protocol it will become feasible to achieve mutual authentication for all data traffic between the network layers of two or more nodes.

2.5.3 MANET Authentication. There has been much research and many solutions proposed in the area of security and authentication for wired and wireless networks [BaB98, KaN93, MTH92, NaS78, PaS98], but little has been done in this area for MANETs. In [ZaH99], Zhou and Haas describe a generic security solution for ad hoc networks using

a public-key cryptosystem and key management service called threshold cryptography within a MANET. Their system distributed the responsibility of creating, maintaining, and distributing key pairs among multiple MANET nodes. The system appears to be an effective solution for key distribution and management; however, its impact on network load, performance, computing requirements, and power efficiency is unknown.

Another authentication scheme was designed by Venkatraman and Agrawal for a MANET using the Cluster Based Routing Protocol (CBRP) [VaA00]. The CBRP architecture is an on-demand routing protocol made up of overlapping or disjoint 2-hop-diameter clusters. Each cluster is identified by its cluster head node. The authentication scheme proposed in [VaA00] uses an unidentified form of public key cryptography so all nodes have a system public/private key pair and a cluster public/private key pair. This algorithm provides the required mutual authentication for CBRP using a sequence of events that relies heavily on the cluster head node to perform the encryption processing work up-front, then distribute the results to the nodes in that cluster. These events incorporate the use of the system key pair, cluster key pair, session key pair, a timestamp, and authentication tags. This research adopted the idea of providing each node with a system public/private key.

A third authentication mechanism is contained in an Internet Engineering Task Force (IETF) Internet draft [JaC99]. This mechanism specifies a scalable MANET Authentication Architecture (MAA) for the Internet MANET Encapsulation Protocol (IMEP). MAA was designed to handle the Secret Key, Message Digest 5 (MD5), Rivest, Shamir, Adleman (RSA), Elliptic Curve (EC) and Digital Signature (DS) encryption algorithms for IMEP messages under most any MANET routing protocol. Among other things, this system generates IMEP authentication and certificate objects that follow all non-authentication

objects. These objects are used with administrator-defined cryptosystems to provide mutual authentication and validation of all MANET routing control messages. In [VaA00], it is recognized that this system would be difficult to implement due to constantly moving nodes and no underlying infrastructure. Thus, it would be difficult to find common certification authorities for any two communicating nodes.

Seung Yi's research [YNK01] seeks to make a security-aware routing protocol for MANETs. This protocol is based on the military concept of ranks and privileges that go with those ranks. This routing protocol proposes a very interesting solution. Data is routing through a specified set of rank(s) of nodes within the MANET based on the security needs of the data being sent.

Lastly, a public key distribution and management system is being developed by Jean-Pierre Hubaux [HBC01] in coordination with the terminodes project [HGB01]. This system seeks to advance research into a more efficient key distribution and management system that can be used by MANETs via public key authentication systems such as being proposed in the research presented here.

2.6 Summary

In providing a general background and literature review for this research, this chapter first presented the current Mobile Ad Hoc Network (MANET) paradigm as described by the Internet Engineering Task Force (IETF) MANET working group. Special emphasis was placed on the Dynamic Source Routing (DSR) protocol as the protocol of choice for this research. Next, authentication mechanisms, developed for wired and wireless networks, were explored to provide an overview of what is currently available to the networking com-

munity. Then, the role encryption mechanisms play in various authentication systems was covered. Lastly, current authentication and security research and mechanisms developed for the MANET paradigm were presented.

III. Methodology

3.1 Problem Definition

In a MANET, there are no central servers or routers from which trusted information can be obtained or that can be used to ensure data is properly routed and received. These functions must be accomplished through the trusted cooperation of nodes within the MANET. However, for MANET nodes to trust other nodes and cooperate with them they must be able to authenticate each other as being valid and trusted nodes. Achieving this authentication for packet transmissions within a MANET is a significant problem. "Good" authentication provides a destination node evidence of a particular secret without the sending node having to reveal the secret [PaS98]. "Mutual" authentication ensures that good authentication is established in both directions, that is, for the sending and receiving nodes. Accepted authentication methods available for fixed network infrastructures often come with the cost of increased bandwidth consumption and computing resource requirements as well as a decrease in network throughput due to larger packet sizes or an increased number of packets. More importantly, most authentication systems rely on trusted central servers which are not available in a MANET. This research addresses the problem of mutual authentication and security within a MANET and the costs associated with incorporating public key cryptography into the Dynamic Source Routing (DSR) protocol.

3.1.1 Goals and Hypothesis. The primary goals of this research are to:

1. Develop an efficient mutual authentication system for a Mobile Ad Hoc Network
2. Determine what performance impact the authentication system has on the MANET

It is hypothesized that incorporating an authentication and security system directly into the routing protocol will result in an efficient method to gain a desired medium to high level of authentication and security while still maintaining an adequate "goodput ratio" as well as acceptable end-to-end delay of packets. "Goodput" ratio is defined as the ratio of data bits successfully received, *dbr*, to all routing overhead bits transmitted, *rbt*, plus the data bits successfully received or $\frac{dbr}{(rbt+dbr)}$. For example, a goodput ratio of 0.5 means that for every 1 data bit received there was 1 bit of overhead information transmitted. This is a "higher is better" metric since as the number of routing bits approaches zero the ratio will approach 1.0.

3.1.2 Approach. To accomplish the stated goals above, the following steps were followed:

1. Select a representative MANET routing protocol. Since there is no standard MANET routing protocol, one was chosen from two experimental standards. The Dynamic Source Routing (DSR) protocol best represents the MANET paradigm for this research since it
   (a) is one of the experimental standards,
   (b) has the ability to support uni-directional links,
   (c) can promiscuously gather network routing information, and

(d) support multiple routes to a destination.

It is unimportant which MANET routing protocol is selected for this research since the baseline performance metrics of the selected protocol will be used as a comparison for the resultant performance metrics of this research. However, DSR seems to be a better match for a targeted military environment compared to other MANET routing protocols.

2. Perform verification and validation. The DSR model for the OPNET network simulation tool, developed by the National Institute of Standards and Technology (NIST) was modified to reflect the Internet Draft specification of the DSR protocol then verified and validated against published performance data as described in Chapter IV.

3. Complete a baseline performance analysis. A simulation of MANET traffic with no authentication mechanisms was developed to establish baseline values for the selected performance metrics. The parameters used for this baseline model are described in Section 3.5.

4. Develop a new protocol with authentication and security built in. Next, a new protocol was developed from the baseline DSR model by incorporating public key cryptography into the DSR processing of routing and data packets. Experiments were conducted with the new protocol using Elliptic Curve Cryptography (ECC) and Rivest, Shamir, Adelman (RSA) and the results were compared to the performance metrics from the baseline experiments.

5. Present analysis and conclusions. After data from these experiments were obtained and analyzed, results and conclusions were laid out in accordance with the goals of this research. The results and interpretation from the comparisons are displayed in a manner sufficient for a reader to comprehend how conclusions were reached.

3.2 System Boundaries

As depicted in Figure 3.1, the System Under Test (SUT) for this study includes all authentication systems involved in a MANET node. This includes the authentication mechanisms and systems used by the applications, the Internet Protocol (IP), the routing protocol, as well as by the Medium Access Control (MAC) protocol. This research produced an authentication system for the MANET routing protocol, hereafter termed the Integrated MANET Mutual Authentication System (IMMAS). Thus, the Component Under Test (CUT) is the IMMAS. Three levels of IMMAS, based on the types of public key cryptography used, are evaluated.

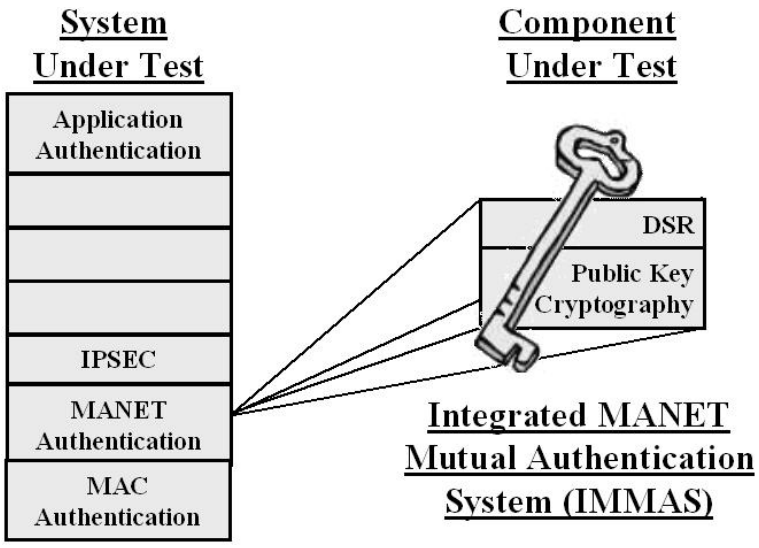

Figure 3.1. Methods of Authentication for Wireless ATM Networks

There are many types of MANETs with nodes ranging in size from small hand-held devices to military HUMVEE communication centers. Each type of MANET has different levels of authentication and security requirements along with very different computation and bandwidth capabilities. For the purposes of this study and in the interest of environment and complexity control, MANET nodes will be homogenous and will have the capabilities of a typical laptop computer with sufficient renewable power resources. This will permit reasonable computation resources for the MANET routing protocol and the IMMAS while providing enough continuous power to permit the maximum transmission and reception ranges.

3.3 System Services

Authentication is one of several services that must be offered by a network for it to be considered secure [HGB01]. IMMAS enables a node to verify the identity of a peer node with whom it is communicating. The primary services and their respective outcomes include:

1. Peer Authentication.
   (a) Success - Packet received from valid network node (packet accepted)
   (b) Failure - Unable to establish that packet came from valid network node (packet dropped)
2. Routing Authentication.
   (a) Success - Packet forwarded by valid network node
   (b) Failure - Unable to establish that packet was forwarded by a valid network node

3. Payload Authentication.

   (a) Success - Packet payload is from valid network node (packet accepted)

   (b) Failure - Unable to establish that packet came from valid network node (packet dropped)

When all three services are successful, any valid node in a MANET is able to communicate with any other peer node within the same MANET. Further, it can trust the routing information and data received as being from a peer node and will thus accept that packet. Without one or more of these authentication services, an adversary could masquerade as a trusted node thus gaining unauthorized access to resources and possibly sensitive information, as well as interfering with the operation of other nodes and the network in general. Not considered in this research are denial of service attacks. The physical security of a node is assumed.

3.4 Performance Metrics

The following metrics were used:

1. Throughput - Throughput is defined as $S = \frac{b_{tx}}{t*N}$, where S is the throughput in bits per second per node, b_{tx} is the number of successfully transmitted bits, t is the observation period, and N is the number of nodes in the MANET. This metric will effectively show the total number of routing and data bits per MANET node being transmitted into the network's "pipeline". This throughput will exclude any MAC-layer bits being transmitted for synchronization, such as RTS/CTS packets used in IEEE 802.11.

Throughput performance is critical since MANETs must operate in a low-bandwidth environment. MANET nodes interfere with each other in an omnidirectional fashion as defined by the power decay law [HGB01]. Thus, for N nodes attempting arbitrary point-to-point communications in a bounded region, the total throughput capacity of the MANET increases by approximately $\sqrt{N}$. This implies that the throughput per node decreases by approximately $\frac{1}{\sqrt{N}}$ [HGB01]. Gupta and Kumar [GaK00] state that the throughput obtainable by each node within a wireless network is $\theta\left(\frac{1}{\sqrt{N \log N}}\right)$ where N is the number of nodes in the network. For this reason the number of nodes in a given range can greatly affect the available bandwidth and services provided by MANET nodes. However, when measuring the throughput of transmitted bits per node there will not be a direct relationship between that throughput and the one described in [GaK00].

2. Goodput Ratio - "Goodput" ratio is defined as $G = \frac{db_{rx}}{rb_{tx}+db_{rx}}$, where G is the ratio, db_{rx} is the total number of data bits successfully received, and rb_{tx} is the total number of routing bits transmitted. For example, a goodput ratio of 0.5 means that for every 1 data bit received there was 1 bit of routing information transmitted. This is a measure of the efficiency of the network and is a "higher is better" metric. Observe that as the number of routing bits approaches zero, the ratio will approach one.

3. End to End Delay - ETE delay is measured in seconds. It is defined to be the elapsed time from when a data packet arrives at the source node's routing layer to when the packet is received by the routing layer of the destination node.

3.5 Parameters

The parameters for this system are numerous, but based on expert knowledge and pilot simulation runs the following parameters are believed to have the largest impact. Pilot studies were conducted on these parameters to ensure they contributed to a change in the performance metrics. Those parameters whose effect was insignificant were dropped from the list. These resulting parameters are grouped by system and workload.

3.5.1 System.

1. Wireless Transmission/Reception Equipment - This equipment can range from directional to omni-directional. This is included as a parameter since omni-directional equipment equates to a greater possibility of more peer nodes within an acceptable area of coverage, each of which are possibly contributing to the volume of data being sent or received by a node at any given point in time. For this research, the equipment used in omni-directional.
2. Data Rate - Typical WLAN data rates range from 1 Mbps to 11 Mbps. This research used a data rate of 2 Mbps.
3. Public Key Cryptosystem - This parameter plays a large role in determining the computing power needed. For instance, RSA takes a lot more computing time and resources when compared to the ECC. This parameter was chosen as a factor.
4. MANET protocol - The network protocol plays a large part in determining the amount of time and overhead resources needed for nodes to determine routes and be able to work with each other. This parameter was chosen as a factor.

5. Simulation Area - The size of the geographical area the MANET performs in contributes to the node degree and plays a part in the decision on how many nodes to place in that area. This research used an area of 1500 x 300 meters for the validation and verification, and an area of 600 x 300 meters for the rest of the research.

6. Cache Size - The size of the cache refers to the number of routes a node's cache will maintain to any particular destination node. Pilot studies showed that a cache of 50 routes to every destination produced the best results under the caching strategy described in Chapter IV.

7. Node Mobility - Node Mobility plays an important role in the randomness and overall outcomes of the MANET performance metrics. This research used the random waypoint model as described in Chapter II.

8. Key Strength - The key strength for the encryption algorithm used contributes to the overall size of the routing and data packets being transmitted. This research used an ECC key strength of 160 bits and an RSA key strength of 1024 bits. These are the lowest acceptable key strengths as defined by [IEE00, IEE01]. In [BSS99], they show that these key sizes produce comparable encryption security.

3.5.2 Workload.

1. Nodes - This parameter is simply the number of nodes used in the research. Fifty nodes were used for this research.

2. Source Nodes - This parameter defines the subset of nodes that originate data packets in a peer-to-peer connection. This parameter was chosen as a factor to vary the

workload and was set at 20 or 30 nodes as per previous research [BMJ98, MBJ99, DPR01]. Pilot runs also showed that if 40 source nodes were used the contention level became too great and the end-to-end delay metric began to show delays of well over 5 seconds in many cases, which was unacceptable for this research.

3. Node Speed - Node Speed is used by the node mobility model to determine how fast a node moves from one point in the geographical area to another. For this research the node speed is uniformly distributed between 0 and 20 meters/second.

4. Node Pause Time - Node Pause Time is also used by the node mobility model. Every time a node reaches its destination the node will wait a PAUSE amount of time before starting to the next destination. This parameter is also used to control the movement of nodes. This parameter was chosen as a factor.

5. Node Degree - The node degree is a measure of the number of nodes that are within the transmission range of any one particular node. Various research including [RSM01, KaS78] has shown that this node degree should be between 6 and 9 neighboring nodes for a stationary network. This provides for minimum partitioning of the network as well as avoiding contention problems experienced at higher node degree levels. Although an optimal node degree for a mobile network has not yet been determined, it is believed that the node degree should not be much higher than that of a stationary network.

6. Transmission Range - The transmission range greatly affects the node degree of a given network. The transmission range was set at 250 meters as per previous research [BMJ98, MBJ99, DPR01].

7. Size of routing and data packets - Routing and data packets must be transmitted among the nodes of a MANET, but doing so can lead to congestion of the limited bandwidth. The amount of data versus overhead in the packets plays an important role in network throughput. This workload is determined primarily by the type of routing protocol used. The data packets were set at 64 bytes as done in previous research [BMJ98].

8. Mean Interarrival Time - The amount of time between packet arrivals was set at 0.25 seconds to provide 4 packets per second as done in previous research [BMJ98]. The workload was changed by varying source nodes versus varying the number of packets generated per source node.

9. Hop Delay - Hop Delay is used to calculate the timing of replies. The specification establishes that this delay should be twice the minimum propagation delay, which is said to be 600 microseconds [MBJ99]. Thus, this delay is set at 1.2 milliseconds.

10. Transmission Delay Window - This delay keeps a node from sending an unbounded number of packets along a route that may be invalid. This is the amount of delay per hop of a source route before a source node sends the next packet to the second node on the route. This delay allows for the reception of an error packet should there be one from the previous packet sent. Pilot runs showed the best value for this delay is 30 milliseconds per hop. For example, if a source node is sending a packet to a destination node that is 5 hops away, the source node (only the source node) will wait 150 milliseconds before sending subsequent data packets on the same route. All other nodes along the route will forward the data packet immediately.

3.6 Factors

The following factors and their corresponding levels were chosen as the most significant for this research based on expert knowledge and pilot studies.

3.6.1 Authentication System.

1. No authentication - This provided a baseline performance analysis of DSR for the other experiments.
2. IMMAS using ECC - IMMAS implemented with Elliptic Curve Cryptography using a key strength of 160 bits.
3. IMMAS using RSA - IMMAS implemented with RSA using a key strength of 1024 bits.

3.6.2 Number of MANET source nodes.

1. Lightly Loaded MANET - 20 source nodes were used to define a lightly loaded MANET.
2. Heavily Loaded MANET - 30 source nodes were used to define a heavily loaded MANET.

3.6.3 MANET node mobility.

1. Low Mobility - Nodes moving with a pause time of 300 seconds.
2. Medium Mobility - Nodes moving with a pause time of 60 seconds.
3. High Mobility - Nodes moving with a pause time of 0 seconds (constant mobility).

3.7 Evaluation Technique

Since MANETs are a new research area, there are few physical implementations available to gather measurements from. Thus, wireless network simulations using OPNET 8.0.C are used. The data was validated against other published implementation data [Per01]. Simulations also provide a controllable environment as well as producing repeatable results.

3.8 Workload

Table 3.1 displays the critical workload parameters and their associated settings for this research. Appendix A outlines the modifications and implementation of the OPNET DSR model for the Verification and Validation Model. Other than the workload parameters in the following table, the models are the same.

Table 3.1. Workload Parameter Settings

Workload Parameter	**Setting**
Nodes in Simulation	50
Source Nodes	20, 30
Data Packet Size	64 Bytes
Mean Interarrival Time	0.25 seconds
Hop Delay	1.2 milliseconds
Packet Send Delay	30 milliseconds
Max Node Speed	20 meters per second
Node Pause Time	0, 60, and 300 seconds
Simulation Area	300 x 600 meters

3.9 Experimental Design

This experimental design consisted of specifying the number of experiments, the factor level combinations for each experiment, and the number of replications of each experiment.

Since this included two factors with three levels and one factor with two levels there were 3 X 2 X 3 = 18 experiments. Thus there were 18 experiments with 5 replications each giving a total of 90 simulations.

3.10 Summary

This chapter described the MANET protocol and authentication system used for this research. The approach by which the authentication system is integrated into the MANET routing protocol as well as how the results are compared were also covered. This chapter described the System Under Test, the Component Under Test, the parameters, workload, and factors that were used during this research.

IV. IMMAS Implementation

In a MANET there are no central servers or routers from which trusted information can be obtained or that can be used to ensure data is properly routed and received. These functions must be accomplished through the cooperation of nodes within the MANET. Mutual authentication for data routed within a MANET is a significant problem. "Good" authentication provides evidence of a particular secret without having to reveal the secret [PaS98]. Mutual authentication ensures this "good" authentication is achieved by both sending and receiving nodes. Consider the scenario in Figure 4.1: Node A wants to send an authenticated message to node D via route A-B-D. Prior to accepting the message, node D must be sure it is truly from node A and has not been tampered with by node B or C. Conversely, node A must also be assured that node D will receive the message unaltered by nodes B or C to achieve mutual authentication between nodes A and D.

To get to node D, the message must pass through node B. For security reasons, node A must be able to authenticate that node B is a valid MANET node and that only node B can route data to D using the source route of A-B-D. Node B must be assured the message received came from node A, that node A is a valid MANET node, and that node A is either the source of the packet or in the source route for the packet.

To accomplish this, nodes A, B, and D must have some secret that can be used to create shared secrets for node pairs (A,D), (A,B), and (B,D) such that only receiver nodes can verify the sender node's secret was used to create the shared secret – all without having to know what the sender node's secret is. This process will provide mutual authentication

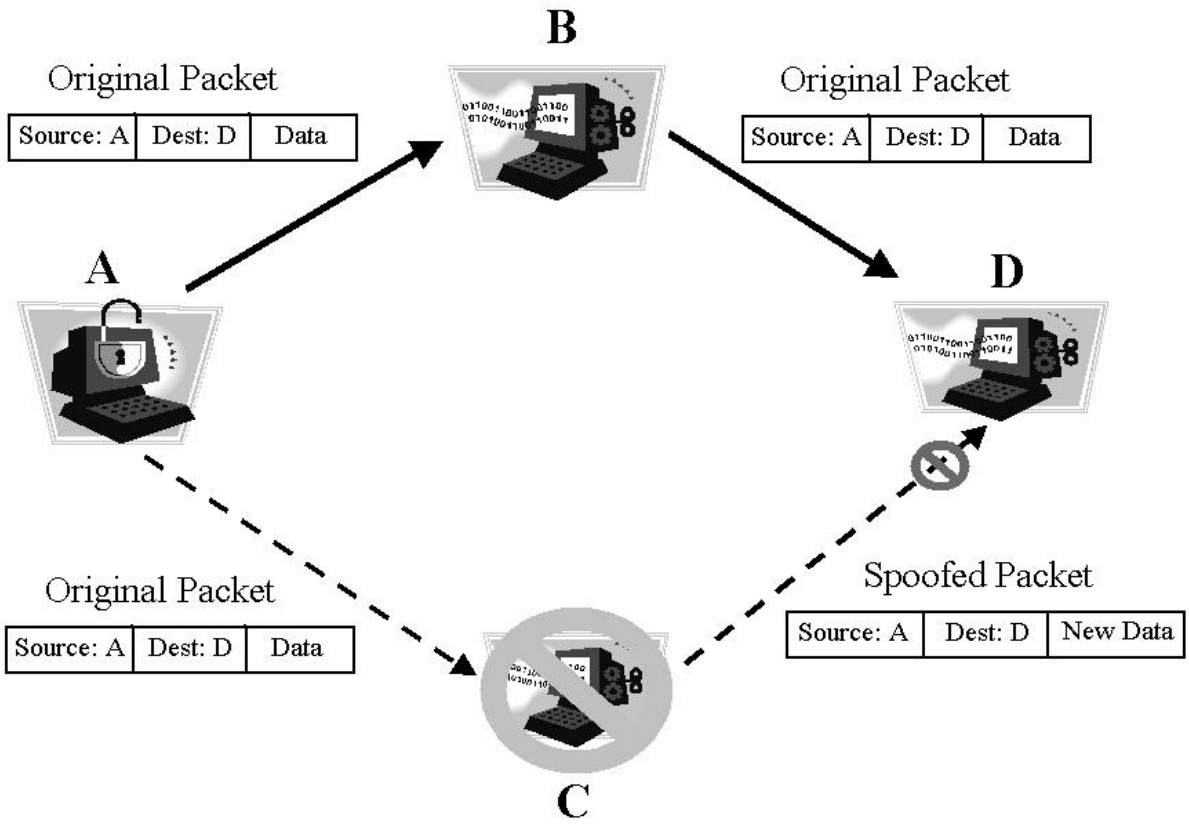

Figure 4.1. Mutual Authentication

between each pair of nodes along the route as well as mutual authentication between the source and destination nodes.

The Integrated MANET Mutual Authentication System is a proposed solution combining the basic ideas of a number of currently available mechanisms and technologies into a new authentication system for MANETs. Some of these technologies include Public Key Cryptography Encryption and Digital Signatures [IEE00, IEE01], Onion Routing [GRS99], Kerberos [KaN93], IPsec [TDG98], and MACKMAN [BaB98]. This system will provide mutual authentication within a MANET by incorporating a public-key cryptosystem called Elliptic Curve Cryptography (ECC) [IEE00, IEE01] into the Dynamic Source

Routing (DSR) protocol. The rest of this chapter lays out how that system is designed as well as how two different cryptographic mechanism can be implemented using IMMAS.

Routing information can be used by an adversary to gather information about the MANET through eavesdropping. This information can also be used for creating a spoof attack or even a man-in-the-middle attack [ZaH99]. Therefore, it is imperative the routing information, whether in routing packets or in the headers of the data packets, be secured such that only nodes belonging to the MANET can read and forward any MANET traffic. However, it is essential for MANET routing, especially DSR, that all packet routing information traversing the network be visible to all nodes belonging to the network. To do this, each node within the MANET is assigned a system public/private key pair along with its own public/private key pair prior to being allowed into the MANET. This will be done with a trusted certificate authority and a secure method of key distribution and management which is assumed to be available for this network.

This system key pair could just as easily be a shared symmetric group key, which would actually decrease the processing, memory, and bandwidth requirements. However, for the ease of comparison, this research used the same cryptographic functions throughout IMMAS. Shared group key(s) are a known security risk since only one node in the MANET needs to be compromised to risk the entire network being attacked. It is assumed that the problem of key distribution and management for MANETs has been solved efficiently and securely to help mitigate this risk by incorporating, among other things, regularly updated group keys. While this only touches the surface of this problem, key distribution and management is beyond the scope of this research. It is a prime area for future research.

In order to achieve the required level of authentication and security, only the source and destination nodes will have access to the plain-text payload data (i.e., all information within the packet following the DSR header). Of course, this is not necessary for DSR generated routing packets as they have no payload data.

Even higher security can be achieved by only allowing those nodes along a specified route to have access to that routing information. However, this effectively removes one of the basic characteristics of the DSR protocol. The DSR specification [JMH01a] permits the source route used in a data packet, the accumulated route record in a Route Request, and the route being returned in a Route Reply to all be cached by any node. While there is a risk of in-the-clear routing information giving an adversary information about the topology and architecture of a particular MANET, it has to be weighed with the fact that passively updated routing information is what makes the DSR protocol more bandwidth efficient.

With this in mind, the packet payload data is encrypted with the private key of the source and the public key of the destination to provide implied mutual authentication and payload confidentiality. This is true as long as a trusted key distribution and management as well as certificate architecture is in place [MOV01, Sch96]. A digital signature is calculated from the entire packet of data by the source node and, in turn, every intermediate node in the source route transmitting the packet. In this way, nodes along the source route are able to verify that the packet came from the appropriate preceding node in the source route and are thus allowed to retransmit a packet. If a packet is transmitted or received by any other node than specified in the source route, that packet will be dropped by all subsequent nodes. This assumes the key distribution architecture is sound and all nodes

that are able to decrypt the routing information are authorized to belong to the MANET and are not "misbehaving". Lastly, the packet routing information, minus the first 4 octets with the routing length information, will be encrypted with the private key of the transmitting node and the system public key. This step ensures only legitimate MANET nodes possessing the system's private key will be able to decrypt the routing information or gain any knowledge about the current routes or topology of the network.

4.1 IMMAS with Elliptic Curve Cryptography (ECC)

To illustrate how IMMAS works, consider Figure 4.2, where Node A wants to transmit a packet to Node D.

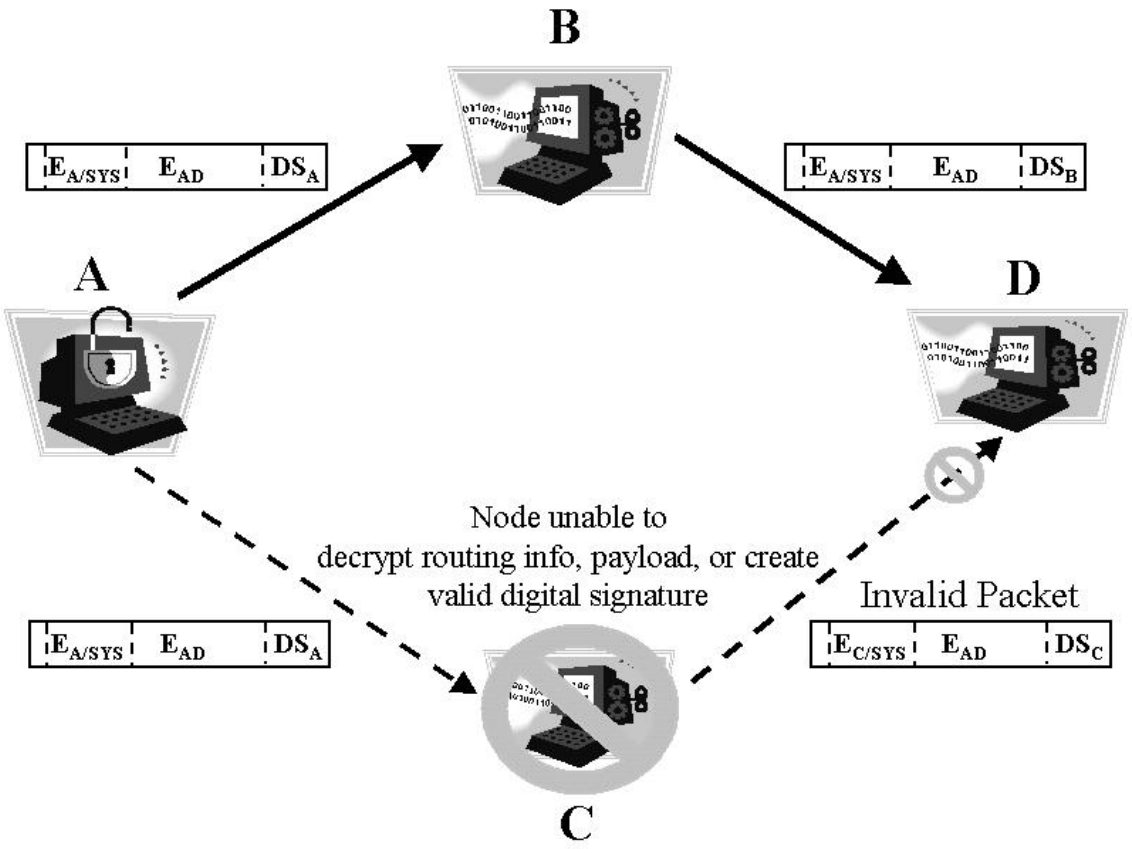

Figure 4.2. IMMAS Implementation

1. Node A uses the Elliptic Curve Integrated Encryption Scheme (ECIES) [IEE01] to encrypt the payload data using its private key and the node D's public key. This is shown as E_{AD} in Figure 4.3. In typical implementations, ECIES would be used to encrypt and decrypt a symmetric session key, which is used to encrypt/decrypt the message or payload. However, the payload and overhead here is small enough that this is an unnecessary additional process.

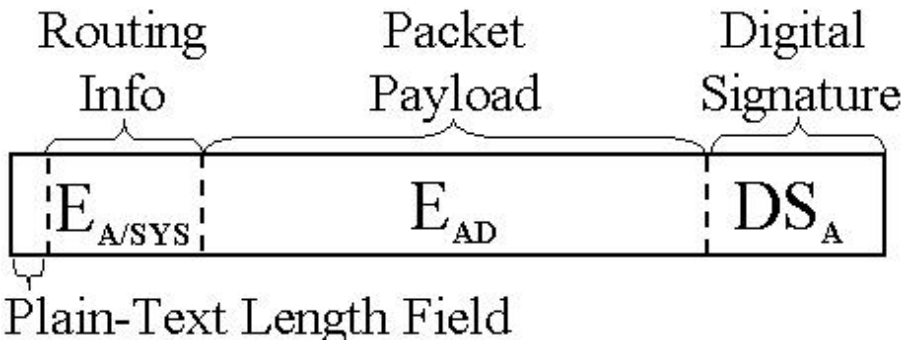

Figure 4.3. Generic IMMAS Packet

2. The Elliptic Curve Signature Scheme with Appendix (ECSSA) [IEE01, IEE00] is applied to the entire packet, which produces Node A's 320-bit digital signature for that packet. This signature is appended to the end of the packet (DS_A).

3. The DSR routing information, except for the first 4 octets, is encrypted with Node A's private key and the system public key ($E_{A/SYS}$). This is a known security risk since every node within a MANET has a copy of both the system private and public keys as well as Node A's public key. Thus, if any node is physically compromised the encrypting of the routing information is rendered pointless. However, since this information still does not decrypt the payload it is a manageable risk.

The first 4 octets of the DSR header remain unencrypted since they contain the payload length field. This information is needed by the receiving nodes to properly decrypt the header. The most significant bit of the DSR packet reserved field could be used to flag when ECC encryption is being used by the MANET routing protocol.

IMMAS allows only nodes belonging to the MANET to decrypt the plaintext routing information. Throughout this process, no node other than the destination can decrypt and read the payload data. As the packet shown in Figure 4.4 is transmitted along the route path from node A to node B the following occurs when it is received.

Figure 4.4. IMMAS Packet Transmitted by Node A

1. The packet routing information is decrypted using node A's public key and the system's private key to gain the plaintext routing information (R), as shown in Figure 4.5. If node B is not in the source route the packet is dropped.

Figure 4.5. Node B Decrypts IMMAS Packet Routing Information

2. The digital signature at the end of the packet is checked to verify that the packet came from node A. If this fails, the packet is dropped and ignored.

3. If the packet header has information pertaining to node B, such as a route request or passing a data packet, it is processed accordingly and the appropriate header fields are updated.

4. Node B produces its digital signature for the packet and overwrites node A's digital signature at the end of the packet as shown in Figure 4.6.

Figure 4.6. Node B Overwrites IMMAS Packet Digital Signature

5. Node B re-encryptes the DSR header, minus the first 4 octets, with its private key and the system public key as shown in Figure 4.7.

Figure 4.7. IMMAS Packet Transmitted by Node B

6. The packet is then transmitted to node D.

Node D will process the packet as follows:

1. The packet is received and the routing information decrypted using node B's public key and the system's private key to gain the plaintext routing information (R) as seen in Figure 4.8. If node D is not in the source route the packet is dropped.

Figure 4.8. Node D Decrypts IMMAS Packet Routing Information

2. The digital signature at the end of the packet is checked to verify that the packet came from node B. If this fails, the packet is dropped and ignored.

3. Since node D is the destination the packet payload is then decrypted with A's public key and D's private key to gain back the original plaintext message (M) as seen in Figure 4.9.

Figure 4.9. Node D Decrypts IMMAS Packet Message

4. The packet is then processed by D accordingly.

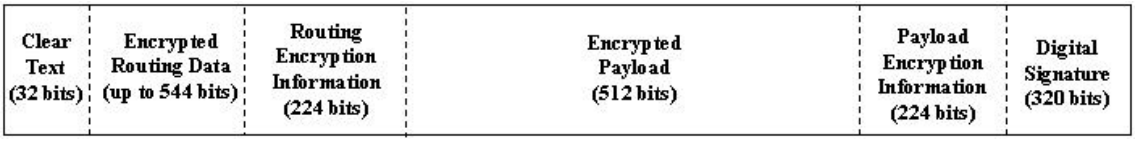

Figure 4.10. IMMAS packet using Elliptic Curve Cryptography

As described above, the 2 encryptions and 1 digital signature will produce a total of 768 bits of overhead – 150% of a 64 byte data packet (refer to Figure 4.10). A DSR routing packet will not have any payload information so only 1 encryption and 1 digital signature will be performed with a total overhead of 544 bits in addition to the actual size of the information within the packet. Using this encryption scheme the size of the DSR headers will approximately double. However, this cost is far below the 1024-bit overhead for each encryption of the RSA algorithm.

Figure 4.11 shows what happens to the size of a data packet before and after a data packet has been encrypted using IMMAS with ECC and RSA. This graph does not include the routing overhead bits that will be added by the routing protocol. For IMMAS using ECC, a total of 768 bits is added to the size of the data packet. For IMMAS with RSA, the final size of the packet follows the formula $P = \left(\left\lceil \frac{M}{696} \right\rceil * 1024\right) + 2048$, where P is the final

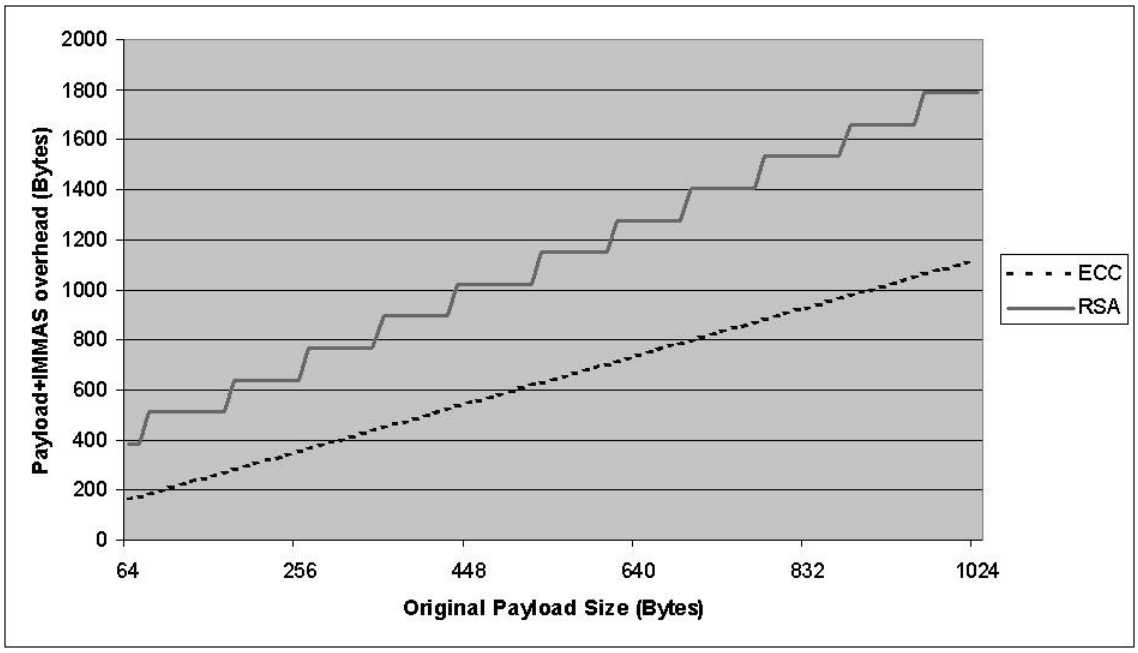

Figure 4.11. Overhead of Two Encryption Algorithms using IMMAS

size of the packet in bits, and M is the size of the original message in bits. The message is divided by 696 bits since this is the max number of bits allowed by [IEE01] for an RSA encryption of 1024-bit strength. The output for each 696-bit piece of the message is 1024 bits, then as shown in Figure 4.12, another 2048 bits is added for the routing encryption and the digital signature.

4.2 IMMAS with Rivest, Shamir, and Adelman (RSA) Cryptography

RSA cryptography is implemented just like the ECC cryptography in IMMAS except that the final size of the fields will be different. According to [IEE00] RSA, with a 1024-bit key strength, will produce 1024 bits for every 696 bits of data to be encrypted. Thus, IMMAS will produce data packets like Figure 4.12 for 512-bit (64-byte) data packets as

was implemented in this research. So in the case of IMMAS using RSA the final size of a data packet will be 3072 bits – a 600% increase in the size of the data packet.

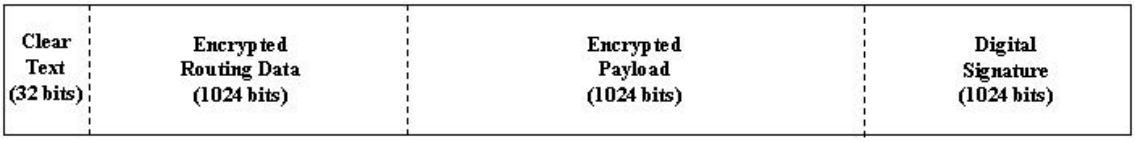

Figure 4.12. IMMAS packet using RSA Cryptography

V. Implementation and Analysis

5.1 Overview

This chapter provides the results of this research's implementations as well as analysis of those results. First, the validation and verification of the OPNET implementation of DSR is described. Second, the implementation used for the baseline DSR model will be described and the results of that implementation shown. Next, the implementation of the IMMAS system will be described using both the ECC and RSA cryptography. Lastly, this chapter will provide an overview of the results and an overall analysis of those results.

5.2 DSR Verification and Validation

In order to verify and validate that the DSR model being used was performing appropriately, simulations were configured and conducted according to previous research in this area [BMJ98, DPR01, MBJ99]. The results were compared to the data provided in that research. In particular, the data provided by [BMJ98] is used as a comparison.

5.2.1 Verification and Validation Implementation. The basic implementation of the DSR model used for verification and validation included the parameter settings defined in Table 5.1.

Performance metrics include the data packet delivery ratio and number of routing packets. More information on the implementation of this verification and validation model is provided in Appendix A.

Table 5.1. Validation and Verification Workload Parameter Settings

Workload Parameter	**Setting**
Nodes in Simulation	50
Source Nodes	20, 30
Data Packet Size	64 Bytes
Mean Interarrival Time	0.25 seconds
Hop Delay	1.2 milliseconds
Packet Send Delay	30 milliseconds
Max Node Speed	20 meters per second
Node Pause Time	0, 30, 60, 120, 300, and 900 seconds
Simulation Area	1500 x 300 meters
Transmission Range	250 meters
Mobility Model	Random Waypoint

5.2.2 Verification and Validation Results. As can be seen in Figure 5.1, all of the data points from previous research implementing DSR in the NS-2 network simulator [Per01, BMJ98] were well above the 97 percent level and the delivery ratios encountered by the OPNET DSR model used in this research all fall above the 97 percent delivery as well. It should be pointed out that the data points from previous research as shown in these graphs are approximate. However, using these data points and assuming a 95 percent confidence interval we find that the two sets of data points are statistically equivalent.

As shown in Figure 5.2, the number of routing packets seen by the OPNET DSR model is statistically equivalent with the data provided by previous research in NS-2. Based on these metrics the OPNET DSR implementation produces similar results to that of previous implementations. Therefore, the OPNET DSR implementation is a valid and verified DSR model.

Figure 5.1. DSR Delivery Ratio Comparison for Validation and Verification

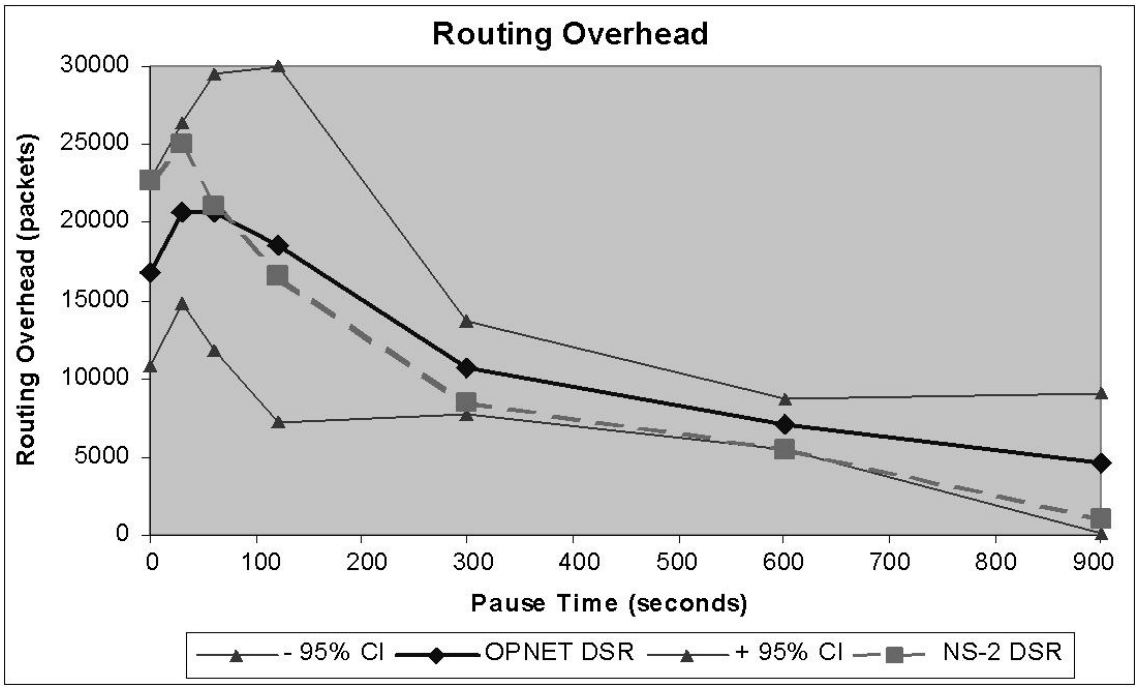

Figure 5.2. DSR Routing Packet Comparison for Validation and Verification

5.3 DSR Baseline

This research is concerned with the effects of the IMMAS system on data efficiency as well as the effects the additional IMMAS overhead bits have on the latency of the network. Thus, in order to gain the desired effects and to create a more "realistic" MANET model for a military environment, the simulation parameters and performance metrics were modified slightly from the verification and validation model described above.

5.3.1 Baseline Implementation. The implementation of the DSR model used as a baseline for IMMAS included the parameter settings outlined in Table 5.2

Table 5.2. DSR Baseline Workload Parameter Settings

Workload Parameter	**Setting**
Nodes in Simulation	50
Source Nodes	20, 30
Data Packet Size	64 Bytes
Mean Interarrival Time	0.25 seconds
Hop Delay	1.2 milliseconds
Packet Send Delay	30 milliseconds
Max Node Speed	20 meters per second
Node Pause Time	0, 60, and 300 seconds
Simulation Area	600 x 300 meters
Transmission Range	250 meters
Mobility Model	Random Waypoint

Performance metrics included the end-to-end delay, transmission throughput, and goodput ratio. Other than the parameter settings described in Table 5.2 and the performance metrics, the baseline DSR model was implemented identical to the validation and verification model described above and in Appendix A.

5.3.2 Baseline Results. Figure 5.3 shows the goodput ratio with an average of 0.607 for 20 sources and 0.597 for 30 sources. In the case of the 20 sources, this means

that for every 1000 data bits successfully received at a destination there were, on average, 647 routing bits transmitted based on the definition given in Section 3.4.

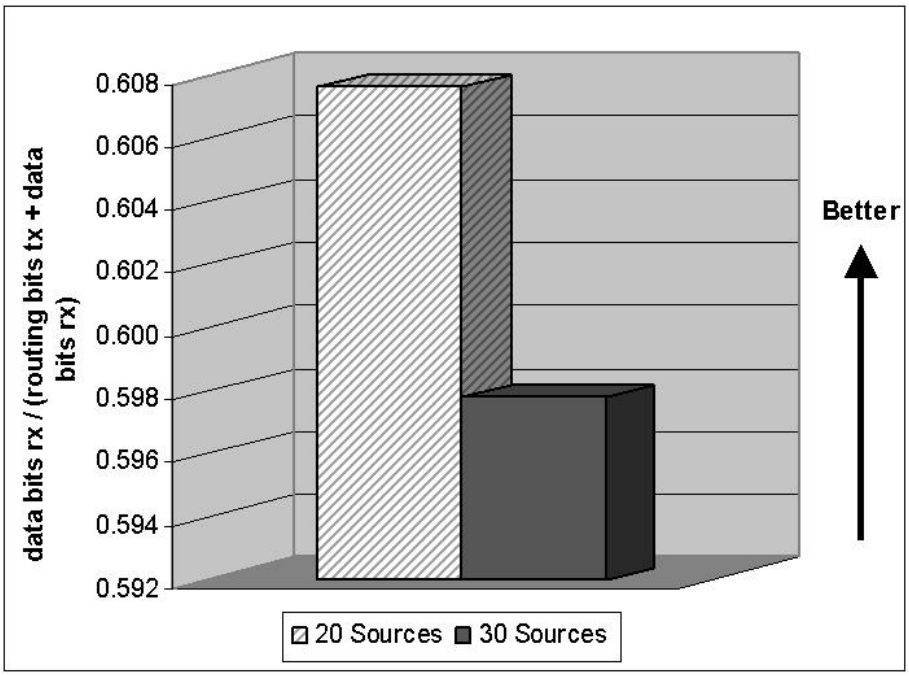

Figure 5.3. Goodput Ratio for OPNET DSR Baseline Evaluation

Each simulation was run for 900 seconds with the source nodes uniformly starting packet generation between 0 and 180 seconds at a rate of 4 packets per second. Thus, for 20 and 30 source nodes there are approximately 65,000 and 98,000 data packets generated respectively. The number of routing packets, shown in Figure 5.4 then proves to be less than 1.5 percent of the total number of data packets in the baseline evaluation for both 20 and 30 sources. The average size of each those routing packets is 320 bits. The average number of routing bits seen in a data packet is 288 bits. The average number of hops taken by a data packet to reach its destination in this network is 1.5 (See Figure 5.5). When this number of bits is multiplied by the average number of hops, then added to the number of

routing packet transmission bits, the total number of routing bits adds up and thus the goodput ratio falls to the level seen in the graph.

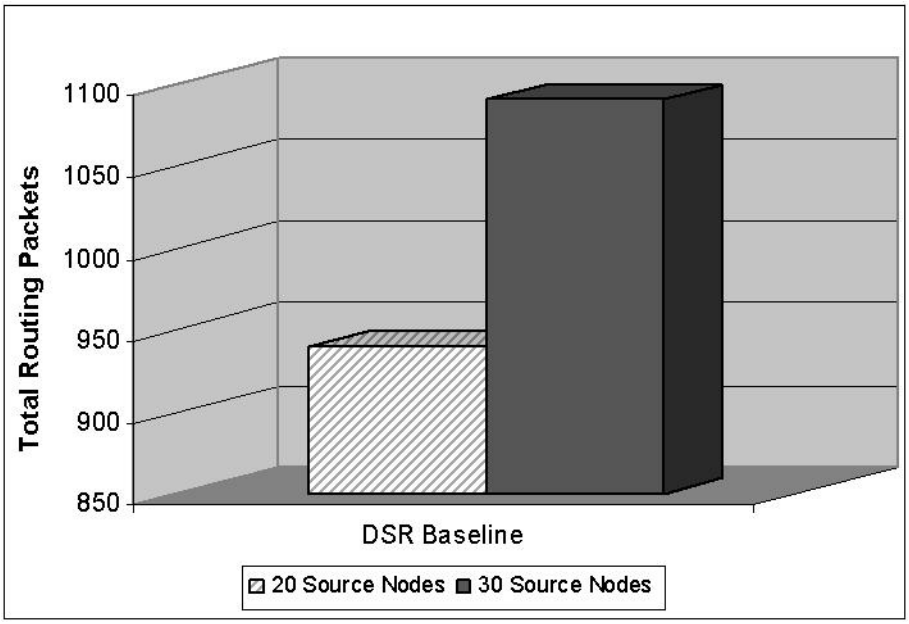

Figure 5.4. Routing Packets for OPNET DSR Baseline Evaluation

For instance, with 1000 data packets of 512 bits (64 bytes) there would be approximately 15 routing packets of 320 bits each required giving 4800 bits. Since there are 288 bits of routing overhead associated with each of the 1000 data packets and each packet took an average of 1.5 hops, there would then be a total of 319,200 bits of routing overhead transmitted with the data packets. The goodput ratio could then be figured by the formula $\frac{512{,}000}{(4{,}800+319{,}200+512{,}000)} = .612$. Thus the goodput ratio would closely resembles the goodput ratio seen for 20 sources. The goodput ratio would, of course, increase under a network passing larger data packets assuming the amount of routing bits would stay the same.

The fact that the average number of hops was 1.5 played a key part in making the movement of the nodes statistically insignificant in the results of all metrics. With the

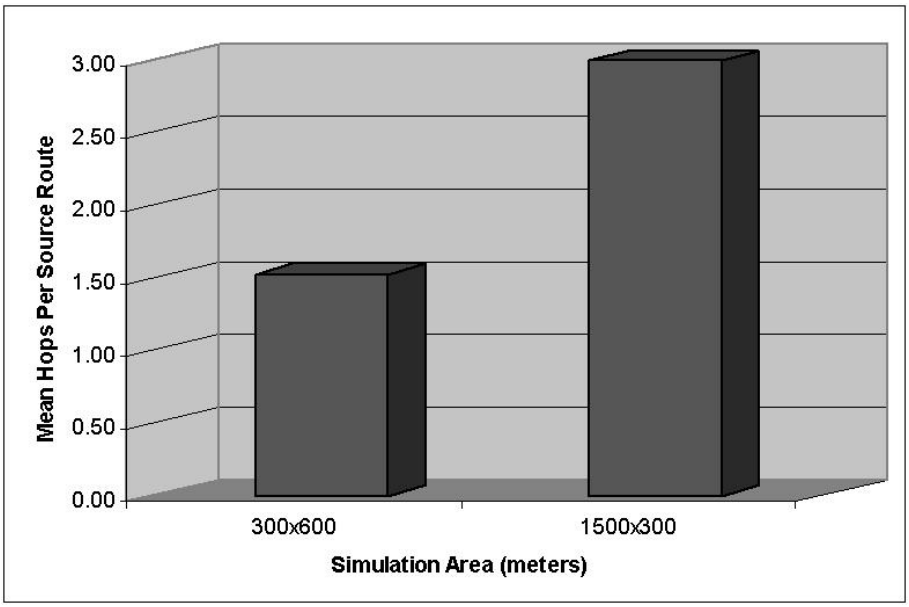

Figure 5.5. Mean Hops Observed Per Source Route

simulation area of 600x300 meters and a transmission range of 250 meters each node will, on average, be in range of 45 other nodes. This means the simulations did not experience the added routing load of route errors due to link breakages and node movement since the nodes can move as much as they want and still be within two hops of any given destination. Thus, the data from each of the pause times was averaged together to produce the metrics for the Baseline, IMMAS with ECC and IMMAS with RSA metrics.

The end-to-end delay seen in this baseline model and shown in Figure 5.6 was minimal. This is due to contention and congestion being at a low level. This level was obtained, in part, by setting the packet size to 64 bytes as well as setting the simulation area and transmission range such that every node can come close to being able to reach every other node. In fact, the maximum number of hops should not be any higher than 3, based on the transmission range of 250 meters and a simulation width of 600 meters.

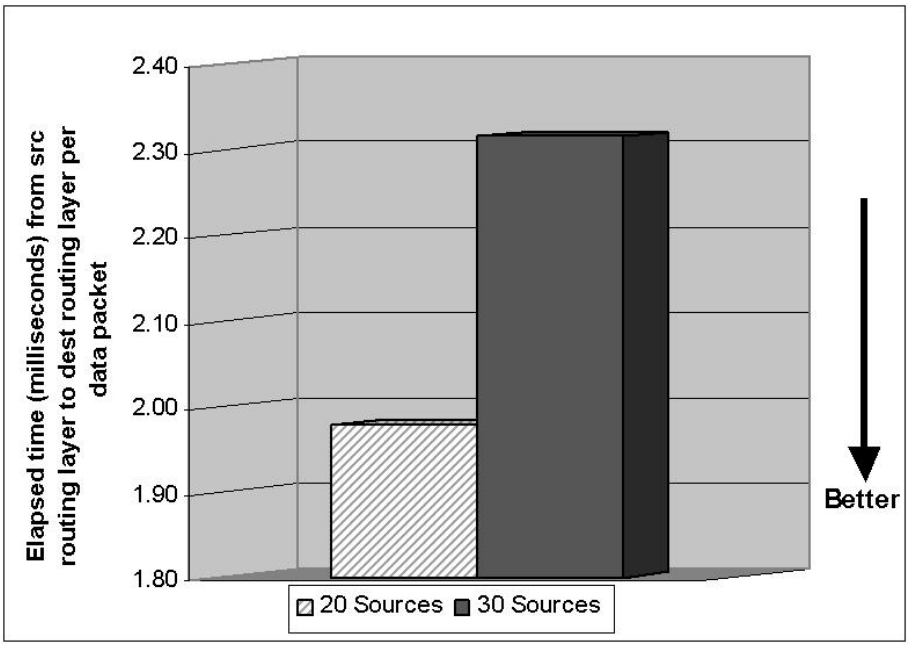

Figure 5.6. End-To-End Delay for OPNET DSR Baseline Evaluation

Any significant increase in the value of these of parameters such as the packet size, transmission range, or simulation area and there will be an increase in the contention and congestion of the network. For instance, some baseline simulations were run using 20 sources with the transmission range set at 100 meters. This change increased the average number of hops to around 3.5, decreased the average goodput ratio to 0.35, increased the average transmission throughput to over 5,000 bits per second and created an average end-to-end delay of over 5 seconds in many cases. Similar, but less drastic, results were seen when the data packet sizes were increased to 512 bytes and the transmission range was left at 250 meters. This was considered unacceptable for this research as the variation for each of these metrics was large and they spoke more about the performance limitations of the routing protocol than about how the IMMAS system would be affecting the network. Unlike other research such as [BMJ98, MBJ99, PRD01], it was not the goal of this research

to stress the routing protocol to determine its breaking points or measure it against other protocols. Thus, for this research the parameters were maintained at the specified levels so the effects of the IMMAS system on the chosen performance metrics could be measured to a greater extent.

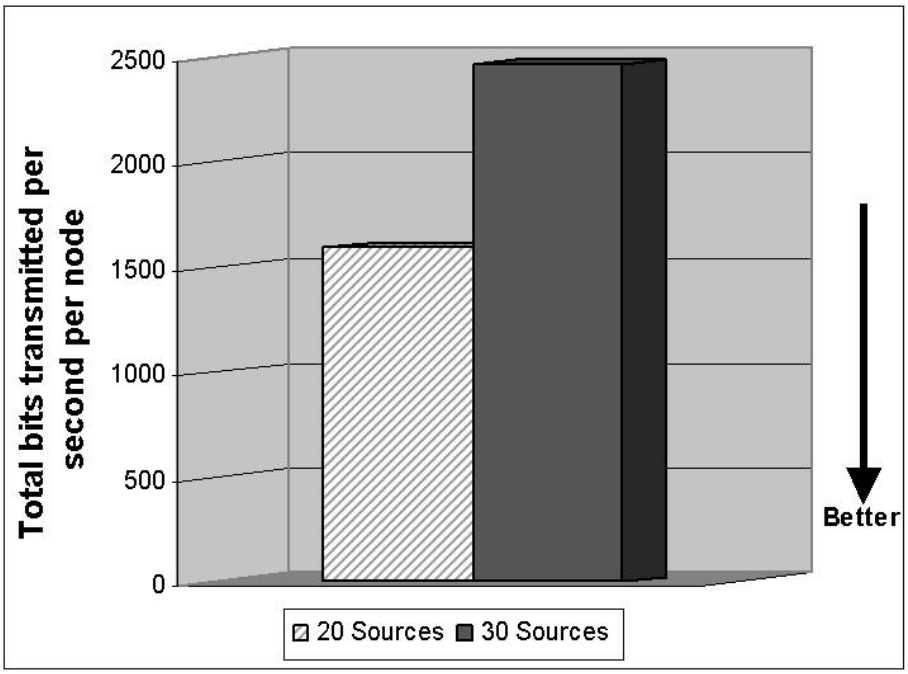

Figure 5.7. Throughput for OPNET DSR Baseline Evaluation

The throughput graph seen in Figure 5.7 shows between 1,500 and 2,500 bits per second per node being transmitted depending on the number of sources. This does not include any of the bits added by the MAC protocol for framing or for synchronizing packets such as RTS/CTS packets, which are considered beyond the scope of this research. Obviously this does not measure the number of bits received by each node, which is dependent on the number of transmitting nodes within reception range. Instead, this metric provides a basis to compare the IMMAS system to.

5.4 IMMAS Implementation

The Integrated MANET Mutual Authentication System, as discussed in detail in Chapter IV, provides the various levels of security shown in Figure 5.3. Each level can be considered optional for implementation based on the assessed security risk of the network. These security levels are all based on the assumption that some type of trusted certificate authority is available for key generations and a secure method of key distribution and management is being applied. These areas will be addressed in future research, but for this research they will be assumed.

Table 5.3. IMMAS Security Options

IMMAS Option	Security Level	Security Effects
No IMMAS System	Low	No Authentication or Security Provided
A	Low/Medium	Payload Security / No Authentication
B	Low/Medium	Peer Authentication / No Security
C	Low/Medium	Routing Security / No Authentication
A & B	Medium	Payload Security and Peer Authentication / No Routing Security
A & C	Medium	Payload and Routing Security / No Authentication
B & C	Medium	Routing Security and Peer Authentication / No Payload Security
A & B & C	Medium/High	Security and Authentication Provided

A = Payload Encryption
B = Digital Signatures
C = Routing Encryption

First, IMMAS uses public key cryptography to encrypt the payload data in a data source route packet at the source node so that only the destination node can decrypt the information. This is achieved by encrypting the payload data with the source private key and the destination public key, thus the payload can only be decrypted using the destination node's private key and the source node's public key. Second, each MANET node will use the information in each packet to create a digital signature that is appended

to the end of every packet. In this way, it can be verified that the packet received came from the appropriate advertised node. Lastly, all routing information in the packet is encrypted with the sending node's private key and the system public key. Any node in the MANET will be able to decrypt this routing information using the sending node's public key and the system private key since every node authorized to be on the MANET will have the system public key and the system private key. Encryption information will be passed with each packet transmission for every encryption completed on the packet, thus overhead is added for the payload encryption, digital signature, as well as the routing encryption when all three options are being used for data packets. For routing packets, only the digital signature and routing encryption overhead is added to the packet since there is no data payload. This research uses the IMMAS system with all options to give a worst-case scenario on results and to provide what is believed to be an adequate level of security.

5.4.1 IMMAS with Elliptic Curve Cryptography (ECC). With IMMAS using ECC for its cryptography, each message encryption carries an overhead penalty of 224 bits and the digital signature consists of 320 bits (Figure 5.8). Therefore, data packets will contain an additional 768 bits of overhead in every packet. For routing packets an additional 544 bits will be added to the overhead of every packet. For more information on the implementation specifics used for these ECC numbers refer to Chapter IV.

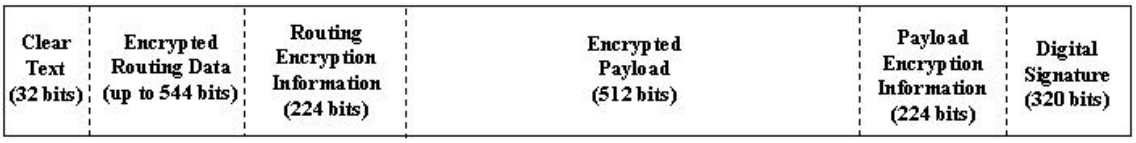

Figure 5.8. IMMAS with ECC Encrypted Data Packet

5.4.1.1 IMMAS with ECC Results. Experiments for IMMAS with ECC were conducted as described in Chapter 3. Simulations included runs varying the mobility rates between pause times of 0, 60, and 300 seconds, and varying the number of source nodes between 20 and 30 sources. Each of these simulations were repeated five times and the averages were used for the graphs presented here.

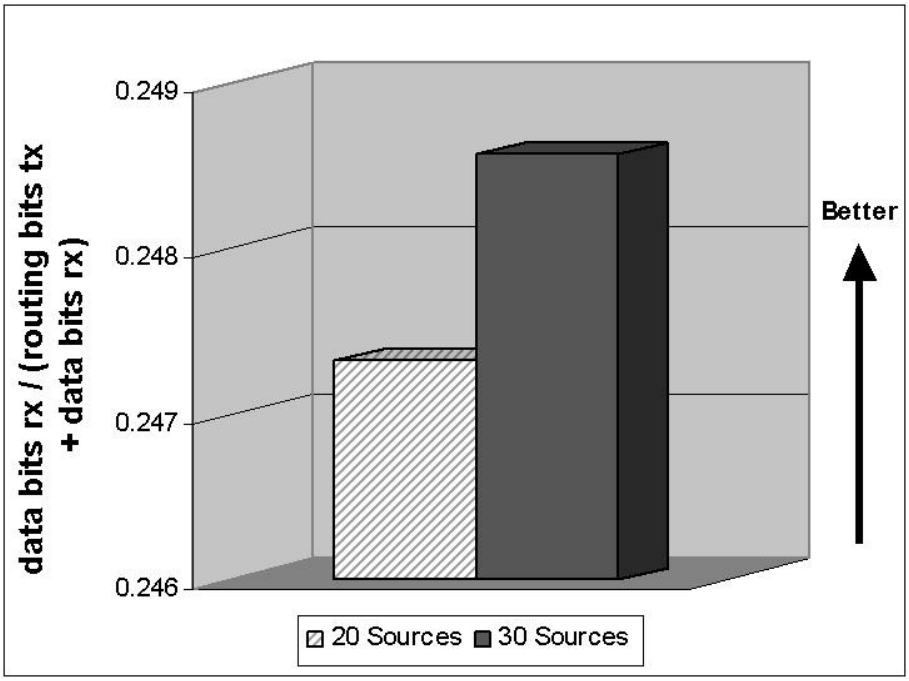

Figure 5.9. Goodput Ratio for DSR IMMAS with ECC

It can be seen from Figures 5.9 through 5.11 that while the goodput ratio for 30 source nodes is only 1.5 percent greater than that of 20 source nodes and the end-to-end delay for 30 source nodes is 4.4 percent greater than for 20 source nodes, the transmission throughput per node shows a full 47 percent increase from 20 to 30 source nodes. This is interesting in that two of the metrics stay nearly the same with an increased load to the network, yet the throughput increased as would be expected under an increased load with more source nodes generating a larger number of data packets going to a larger number

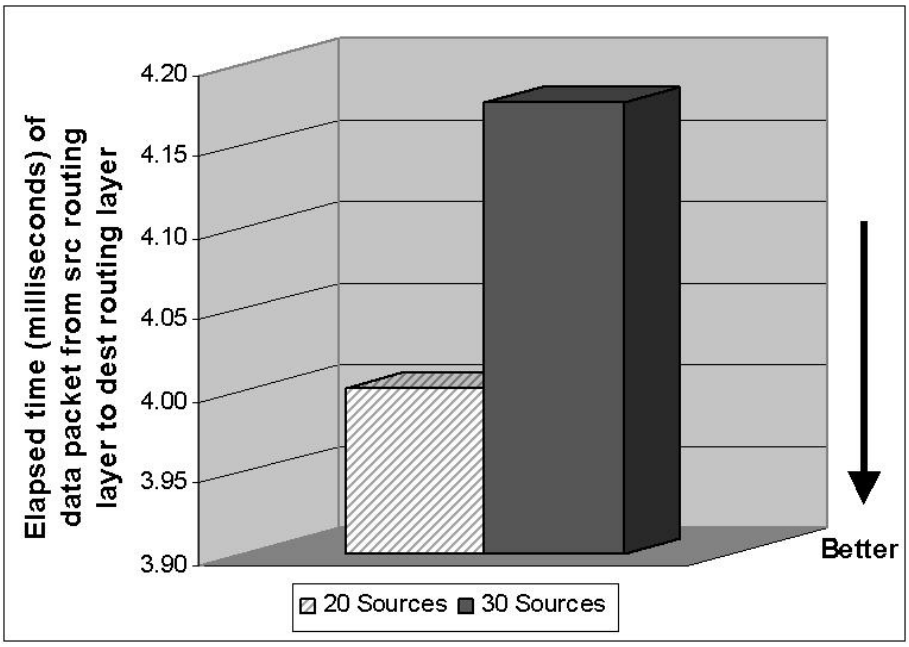

Figure 5.10. End-To-End Delay for DSR IMMAS with ECC

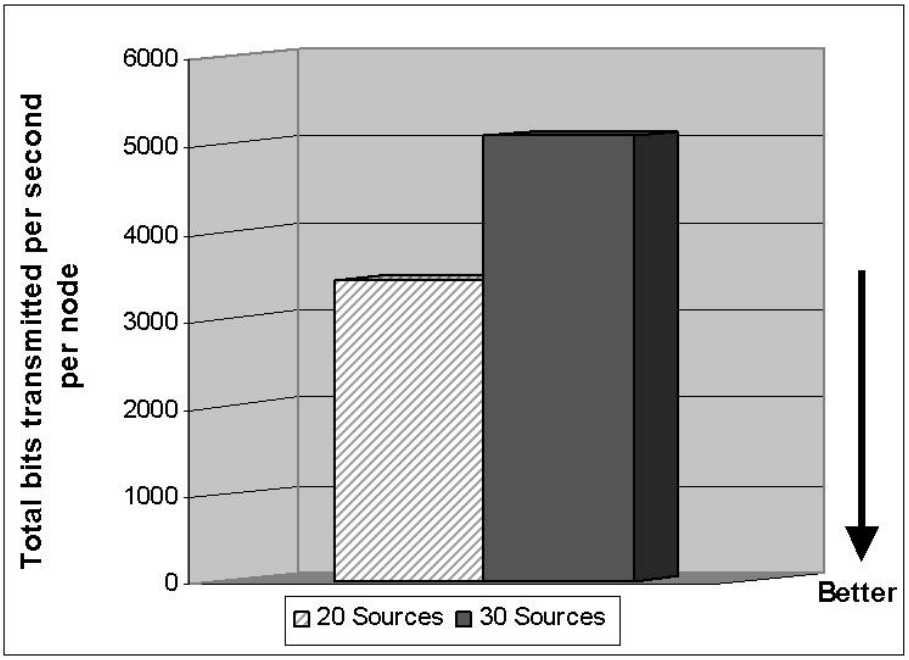

Figure 5.11. Throughput for DSR IMMAS with ECC

of destinations. However, this can be explained by the fact that the amount of overhead added to each packet stays nearly constant with the increased load of 30 sources which

keeps the goodput ratio relatively constant with a difference of 0.001. An increased network load would also normally contribute to a higher ETE delay, but since the destinations can be reached in an average of 1.5 hops and the load to the network is not high enough to cause significant contention, the ETE delay rises only slightly. The variation is statistically insignificant (refer to Appendices B through D for statistical tables). Section 5.5 compares the metrics seen in this section to the metrics seen in the baseline evaluation and the metrics gathered from the IMMAS with RSA experiments.

5.4.2 IMMAS with Rivest, Shamir, and Adelman (RSA) Cryptography. With IMMAS using RSA for its cryptography, each message encryption creates 1024 bits for every 696 bits of the message to be encrypted; the digital signature and the encrypted routing data also consists of 1024 bits. So for data packets with a payload of 696 or less bits (87 bytes), as shown in Figure 5.12, the total size of the transmitted data packet will be 3072 bits (384 bytes). For routing packets, the total packet size will be 2048 bits (256 bytes). For more information on the implementation specifics used for these RSA numbers refer to Chapter IV.

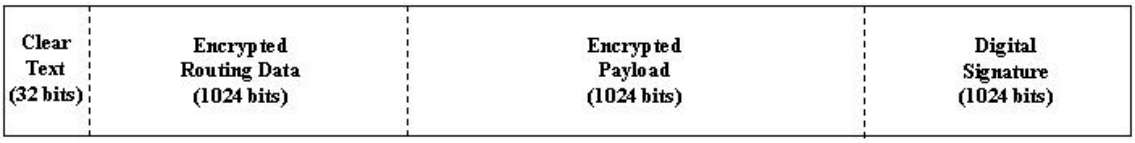

Figure 5.12. IMMAS with RSA Encrypted Data Packet

5.4.2.1 IMMAS with RSA Results. Experiments for IMMAS with RSA were conducted as described in Chapter III. Simulations included runs varying the mobility rates between pause times of 0, 60, and 300 seconds, and varying the number of source

nodes between 20 and 30 sources. Each of these simulations were repeated five times and the averages were used for the graphs presented here.

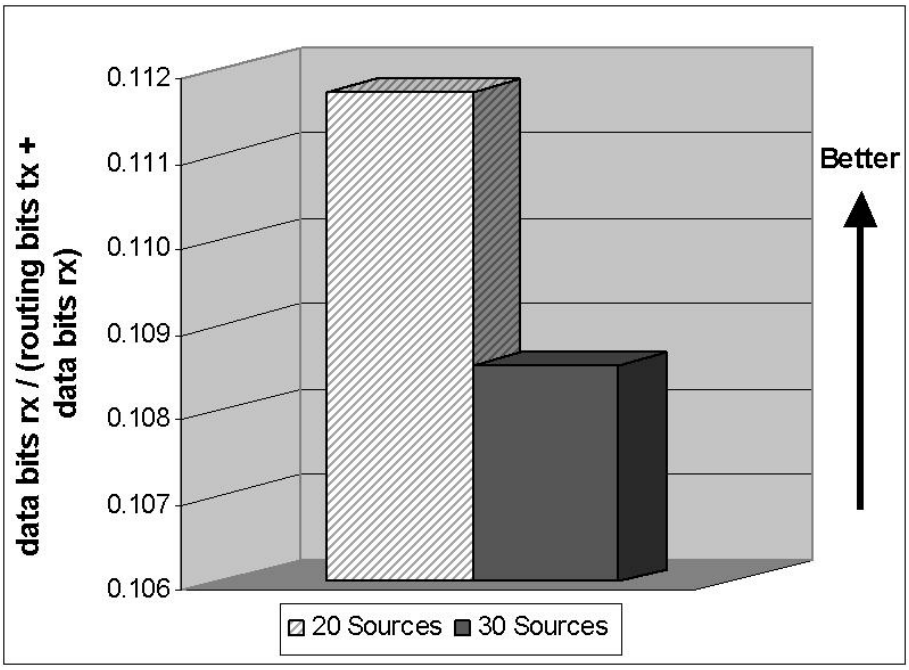

Figure 5.13. Goodput Ratio for DSR IMMAS with RSA

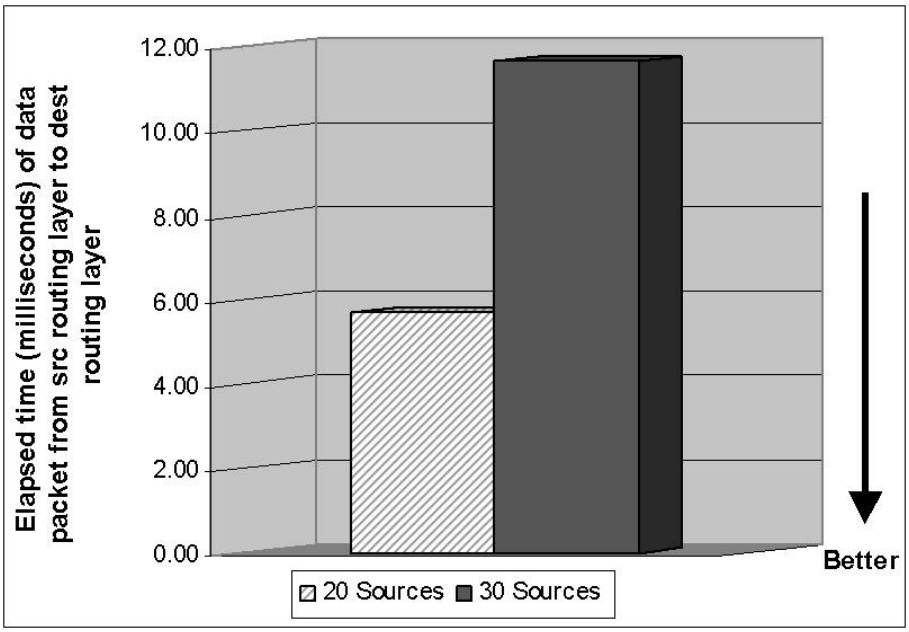

Figure 5.14. End-To-End Delay for DSR IMMAS with RSA

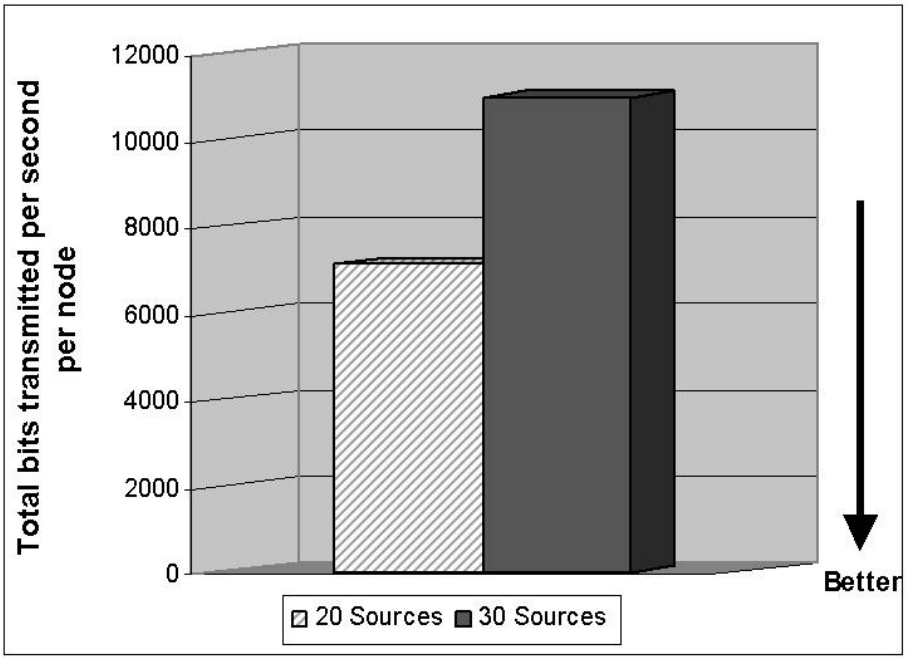

Figure 5.15. Throughput for DSR IMMAS with RSA

The goodput ratio for 20 source nodes is only 2.9 percent greater than that of 30 source nodes. Although the IMMAS with ECC goodput ratio for 30 source nodes is greater than with 20 source nodes, the differences between the goodput ratios is considered insignificant in both systems as they only differ by 0.001 and 0.003. Thus the goodput ratio for IMMAS with RSA is similar to what was seen in the IMMAS with ECC goodput ratio in that the goodput ratio is nearly constant between the two levels of source nodes for each IMMAS system.

On the other hand, the ETE delay for 30 source nodes is 104.9 percent greater than 20 source nodes and the transmission throughput showed a 54.6 percent increase from 20 to 30 source nodes. The increase in these last two parameters is reasonable since IMMAS with RSA is creating a heavier load to the network due to the greater amount of overhead compared to what was seen in IMMAS with ECC. Also as seen with IMMAS using ECC,

with more source nodes there will be a larger number of data packets generated going to a larger number of destinations, which accounts for the increase in the throughput. However, with the increase of possible destinations there will also be an increase in the routing load as the nodes maintain routes to the destinations. This would also normally contribute to a higher end-to-end delay, but since the destinations can be reached in an average of 1.5 hops and the load to the network is still not high enough to cause contention, the end-to-end delay rises slightly but remains statistically insignificant (refer to Appendix B for statistical tables).

5.5 Result Analysis

The next few sections will present an analysis comparing the three levels of authentication for the goodput ratio, end-to-end delay and transmission throughput performance metrics respectively. First, however, Figure 5.16 shows the number of routing packets that were observed for each of the authentication systems for 20 and 30 source nodes respectively. This information simply provides an idea as to the routing packet load for the scenario used in this research which varies greatly under other simulation parameter settings. As discussed in Section 5.3.2, the observed data stayed statistically the same across the different mobility settings of 0, 60, and 300 second pause times and were thus averaged together for the 20 and 30 source node data shown. The Delivery Ratio for each of the scenarios was above 99 percent, so it was not used as a discriminator.

5.5.1 Goodput Ratio Analysis. As seen in Figure 5.17, the goodput ratio follows a downward trend from the baseline to the IMMAS with ECC to the IMMAS with RSA.

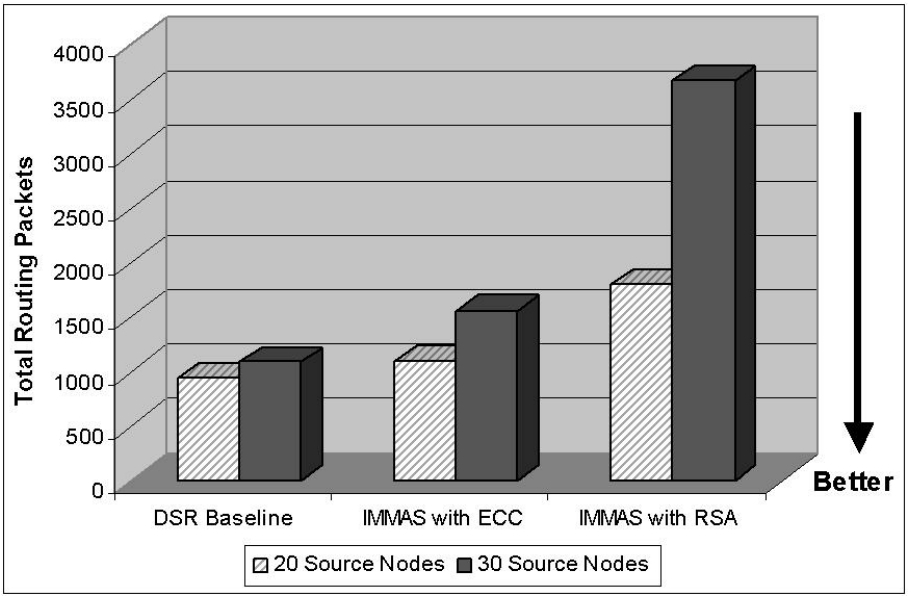

Figure 5.16. Routing Packet Comparison between IMMAS Systems

This was to be expected based on the amount of overhead added by ECC and RSA as described in Chapter IV. The interesting information is how much the IMMAS system degrades the goodput ratio using ECC versus RSA cryptography. The IMMAS with ECC goodput ratio is approximately 60 percent less (approximately 2/5 of the baseline) than the baseline for both 20 and 30 sources. The IMMAS with RSA goodput ratio is as much as 82 percent less (approximately 1/5 of the baseline) than the baseline for both 20 and 30 sources. This is a difference of 22 percent between the systems when they are compared to the baseline system. Thus the goodput ratio for the IMMAS with RSA system is 22 percent worse than the IMMAS with ECC system.

The analysis of variation for the goodput ratio in Table 5.4 shows that almost all the variation, 99.36 percent, is due to the authentication systems. This suggests that values shown in Figure 5.17 are in fact due to the authentication system and not any of the other varied parameters. The goodput ratio of any MANET will vary based on the average

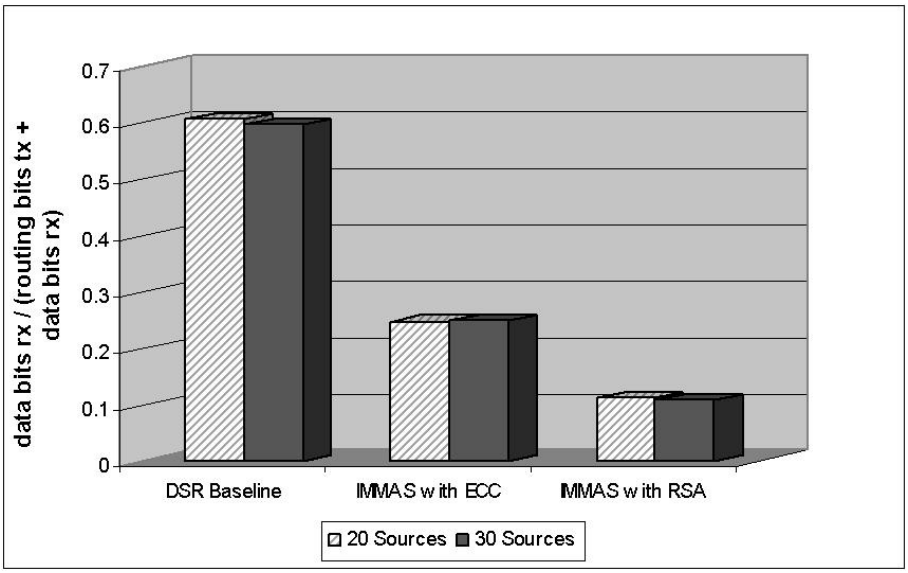

Figure 5.17. Comparison of Goodput Ratios

number of hops taken by the data packets as well as the data packet size. In this scenario, the average number of hops is only 1.5 and a small data packet size of 64 bytes provides a high goodput ratio. An area of future work would be to determine how variations of these parameters affect the performance of the network.

Table 5.4. ANOVA on Goodput Ratios

Source Nodes	Authentication System	Pause Time	Source Nodes and Authentication System	Source Nodes and Pause Time	Authentication and Pause Time	All Factors	Error
0.01%	99.36%	0.01%	0.01%	0.01%	0.01%	0.04%	0.55%

5.5.2 End-To-End Delay Analysis. The end-to-end delays shown in Figure 5.18 produced the expected trend of longer delays with the larger data packets. The variation, in most cases, was not as drastic as seen in the goodput ratios. The IMMAS with ECC produced a 102 percent and 80 percent longer delay compared to the baseline, while the IMMAS with RSA system produced an 187 percent and 403 percent longer delay for

20 and 30 sources respectively when compared to the baseline. This is a difference of approximately 86 percent between the 20 sources of IMMAS with ECC and IMMAS with RSA when compared to the baseline. The difference between the two systems then greatly increases under the heavier workload of 30 sources producing a difference of 323 percent between the two systems when each are compared to the baseline. The jump in the end-to-end delay for the RSA system with 30 source nodes is due to the rising contention levels of the network as the size and number of the data packets gets larger. As discussed in section 5.3.2, the decrease in transmission range or increase in packet size begins to greatly affect the end-to-end delay. These experiments were kept below that point of saturation as much as possible.

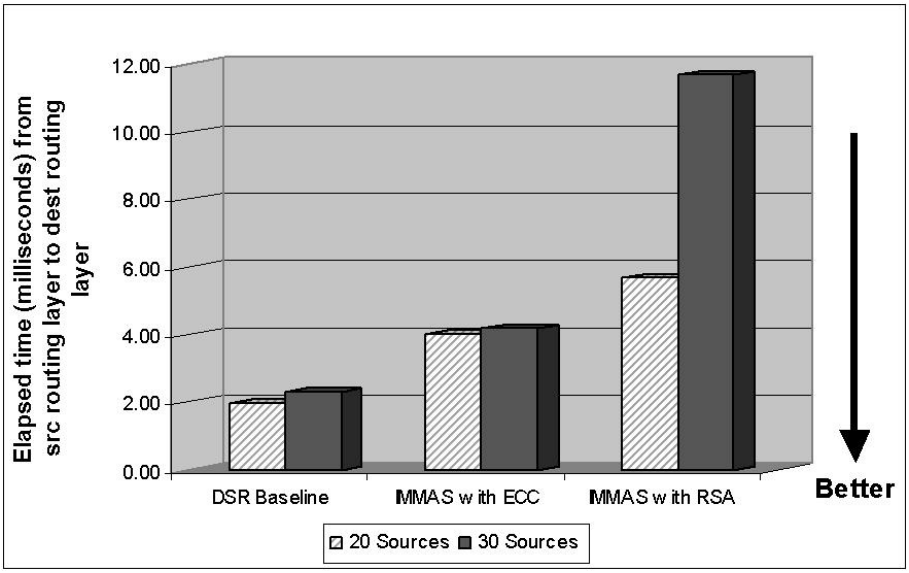

Figure 5.18. Comparison of End-To-End Delays

Like the goodput metric, the analysis of variance for the ETE delay metric in Table 5.5, shows the largest amount of variation, 99.21 percent, is due to the authentication system.

Table 5.5. ANOVA on End-To-End Delay

Source Nodes	Authentication System	Pause Time	Source Nodes and Authentication System	Source Nodes and Pause Time	Authentication and Pause Time	Due to All Factors	Due to Error
0.01%	99.21%	0.01%	0.01%	0.01%	0.01%	0.05%	0.69%

5.5.3 Transmission Throughput Analysis. The transmission throughput data, as seen in Figure 5.19, was similar to the end-to-end results. The IMMAS with ECC system produced 116 percent and 106 percent more bits to be transmitted compared to the DSR baseline. The IMMAS with RSA system produced an average 345 percent more bits to be transmitted for both 20 and 30 source nodes when compared to the DSR baseline system. Thus, IMMAS with RSA produces 234 percent more total bits than the IMMAS with ECC system does when they are compared to the baseline system. These results follow an expected trend as the number and size of the data packets get larger.

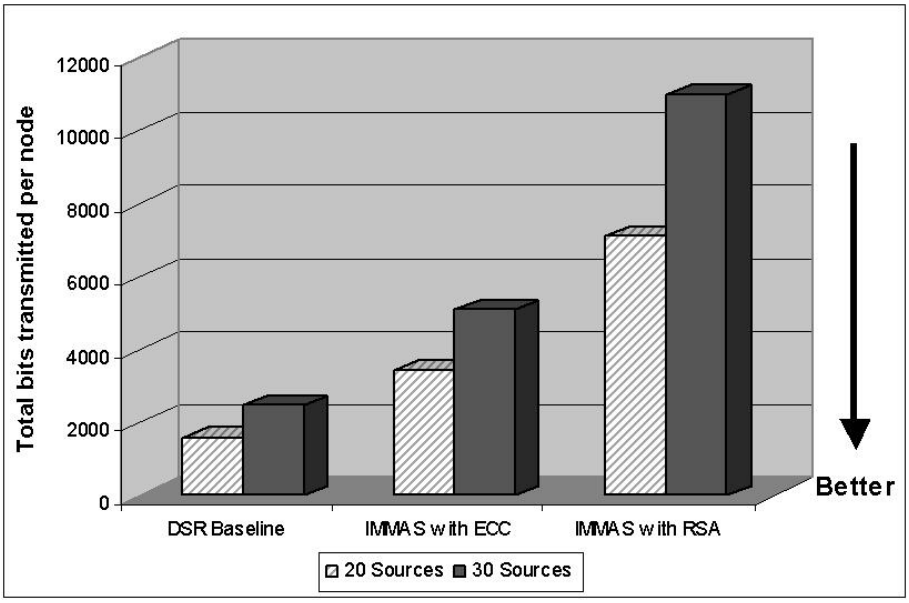

Figure 5.19. Comparison of Transmission Throughput

Once again, at 82.8 percent of the of the variation allocated to the authentication systems, the systems being tested were being appropriately measured. This is shown in Table 5.6.

Table 5.6. ANOVA on Transmission Throughput

Source Nodes	Authentication System	Pause Time	Source Nodes and Authentication System	Source Nodes and Pause Time	Authentication and Pause Time	All Factors	Error
10.96%	82.80%	0.01%	3.94%	3.00%	0.03%	0.09%	2.13%

5.5.4 Conclusions. Overall, the results of this research followed the predictable path of the IMMAS with ECC system degrading the performance of the network, but the IMMAS with RSA increased the network's end-to-end delay and transmission throughput by as much as 323 percent and the goodput ratio was decreased by approximately 22 percent compared to the IMMAS with ECC system. It is a known fact that security comes at a cost and these results show what that cost is for medium-high security.

5.6 Summary

In summary, this chapter described the implementation of the verification and validation model used for the DSR protocol and the resultant data from that model. Next, the implementation and results of the DSR baseline model used for this research were explained. Then a performance summary of the IMMAS implementation using ECC and RSA was provided to display the associated costs of using those cryptography systems. An analysis was performed on those results and the amount of impact the IMMAS system has on a MANET was measured. By performing this analysis it was shown that the medium-

high level of security obtained from the IMMAS system for MANET packet transmissions comes at a cost that must be minimized.

VI. Conclusions and Future Work

6.1 Overview

Security and authentication of transmitted data within a MANET are of utmost importance. Elliptic curve cryptography integrated into the MANET routing protocol provides an efficient means to produce the required level of security and authentication desired by most any mobile organization. IMMAS provides mutual authentication and security while not overtaxing the processing or bandwidth capabilities of a wireless network. In addition to mutual authentication, IMMAS also provides multi-level encryption which ensures the integrity, confidentiality, and non-repudiation for data packets every step along the way. The security provided by IMMAS makes an effective and efficient use of DSR routing and Elliptic Curve Cryptography. Thus, the goals of this research were met by the development of the IMMAS system and the results presented in Chapter V from the simulation of IMMAS.

6.2 IMMAS Conclusions

It can be concluded that the development of the IMMAS system, while not dependent on a particular type of encryption, was shown to be much more efficient using ECC than RSA. Not only was RSA more costly in terms of larger packet sizes and overhead, but it is known to take as much as ten times longer to process compared to ECC [BSS99]. By incorporating authentication and security into a MANET routing protocol this system becomes the first known system of its kind. Other systems provide add-on security, and many only authenticate and protect the the routing data. This system provides security for

both the routing data and the payload, and it touts graceful degradation of security should one aspect of its security shell be compromised. Thus it can be surmised that IMMAS provides the best known security to date for MANETs. IMMAS using ECC provides the same level of authentication and security with the least expensive cost for this security compared to IMMAS using RSA. It is also believed that IMMAS with ECC provides the greatest amount of authentication and security at the least cost to the MANET compared to any other form of cryptography.

6.3 DSR Conclusions

In the efforts of this research to study the effects of the IMMAS system on a MANET using the DSR routing protocol a number of conclusions about the implementation of DSR were reached. These conclusions were not incorporated into the NIST DSR model and thus make a significant contribution to future research in this area. These conclusions are discussed in more detail in Appendix A, but the most critical of these conclusions include the following.

1. Route Cache. Multiple routes to a destination are essential for DSR. If only one route is maintained and that route "breaks" then a new route discovery sequence will have to be initiated. If routes are cached, the node can simply look into its route cache for the next available route. Not caching routes will increase the number of routing packets by as much as two and three times, which also increases the load of the network, the end-to-end delay, and throughput.

2. Packet Salvaging. Packet salvaging should be used extensively to not only increase the packet delivery ratio, but to also clean out invalid routes from the route cache of all neighboring nodes.

3. Data Packet Transmission Delay Window. If multiple packets are waiting in the send buffer for a particular destination, the node should wait 30 milliseconds per hop after receiving a successful acknowledgement from the next hop in the source route before sending the next data packet down that same route. This allowed enough time for the data packet to be transmitted to the destination and for an error packet to make it back to the source node should an error occur.

The research done on the OPNET DSR model has now provided the first DSR model to be implemented and verified and validated against not only the DSR specification, but also against previous implementations of DSR in other simulation platforms. This model will be submitted to the IETF MANET working group for other OPNET researchers to use and expand on.

6.4 Contributions

This research lays the groundwork for authentication and security being built into a MANET routing protocol by providing a quantitative analysis on one particular method. IMMAS goes a long way toward providing current and future researchers of MANETs the tools and data necessary to conduct quantitative and qualitative research without having to assume that authentication and security are present with little or no effect to the MANET performance. IMMAS itself provides graceful degradation of authentication and security that can be built upon and possibly improved with further research.

Furthermore, this research provides a number of conclusions and contributions for the DSR protocol. It has provided the first validated DSR model to the MANET community using the OPNET network simulator and provides yet another avenue to explore this ever-evolving paradigm of mobile ad hoc networks. The DSR model has been provided to the National Institute of Standards and Technology and to OPNET to allow other researchers easy access to the model. The lessons learned and documented in the conclusions will provide an invaluable reference to others.

6.5 Future Work

Future research needs to be done in four broad areas.

1. Key distribution and management for MANETs,
2. Encryption processing requirements for MANETs,
3. Testing other authentication systems using DSR, and
4. Researching the effects of IMMAS with other MANET routing protocols.

Key distribution and management is a very large problem that has the attention of many researchers, and is still presenting an almost crippling overhead cost to wireless networks. While there has been some research in this area, little has been published and accepted as a standard for MANETs. The research and ideas surrounding wireless public key infrastructures need to be extended into the realm of MANETs.

Very little literature was found on the processing requirements for elliptic curve cryptography in MANET environments. This should be studied to determine those effects. It has been stated that if ECC is performed by every node for a route of 10 hops, the

increase on the end-to-end delay could be as much as 0.1 seconds [Ano02]. However, this claim is unverified but should be looked at in future research.

As the IETF MANET working group and industry come to a consensus on standard protocols for MANETs, this IMMAS authentication system should be ported into those protocols and tested for viability. It is entirely possible that under some scenarios and protocols, the IMMAS system could be too much for the MANET. In that case, the MANET administrator(s) have to determine where they stand on the trade-off of network size and/or performance versus security.

Appendix A. Dynamic Source Routing Protocol Verification and Validation Implementation

A.1 Overview

This appendix presents the design and implementation of the Dynamic Source Routing (DSR) protocol. The first area discussed is the modifications and additions made to the OPNET DSR model implementation from the National Institute for Standards and Technology (NIST) [PRP00]. Second, a list of system parameter settings will be described and explained. A list of the workload parameter settings will also be described and explained. Finally, some of the problem areas encountered while implementing this verification and validation model will be discussed.

A.2 Validation and Verification of the OPNET DSR Model

The first step in the process of this research was to develop a model that accurately represented the DSR protocol. The OPNET network simulation tool was chosen for this research. Since NIST had developed a DSR model in OPNET [PRP00], it was chosen as a starting point for this research. The NIST model [JMH99], the third version of the specification for DSR. A number of critical areas in the model were either implemented incorrectly or left out altogether. The NIST model was updated to DSR specification version 5 [JMH01a] so that the model would reflect a current DSR protocol for MANETs. The following list of areas were either modified or added to the NIST model. The only capabilities that were not implemented were the piggybacking of multiple packets into one

packet and the Implicit Flow State for DSR (which now has its own specification separate from the DSR specification).

1. Packet Sizes and Formats. The packet formats and field sizes were not in compliance with the specification for DSR. For instance, all of the address fields were 8 bits instead of 32. While this was all that was needed to hold the address for simulations, it is not the true size of the fields and will impact the load to the network.

2. Route Cache. The NIST model only implemented a single route to every destination. The route cache, as defined in the specification, should allow for more than one route to a destination. While modifying the NIST model for this validation and verification it was discovered that the route cache and the route caching strategy have a large effect on the performance of the network. If only one route is maintained and that route "breaks" then a new route discovery sequence will have to be initiated. If routes are cached, the node can simply look into its route cache for the next available route. Not caching routes proved to increase the number of routing packets by as much as two and three times, which also obviously increases the load of the network, the end-to-end delay, throughput and so on. Therefore, the route cache was modified to handle up to 100 routes per destination with a caching strategy that prioritizes the route based on when the route was added to the cache, the size of the route, as well as how the route was discovered.

3. Packet Salvaging. Packet Salvaging, as defined in [JMH01a], allows for a intermediate node to look for an alternate route in its cache to a particular destination if an error was received for the source route defined in the data packet. This was not

implemented in the NIST model. Packet salvaging is used extensively to not only increase the packet delivery ratio, but to also clean out invalid routes from the route cache of all neighboring nodes.

4. Error Packet Handling. Upon receiving an error for a particular data packet an intermediate node would transmit an error packet back to the source along the reverse path of the source route. However, only nodes in the reverse path would clean out their cache from the invalid link, leaving neighboring nodes with this erroneous route information to possibly use in the next route discovery. This was modified to meet the specification such that all nodes overhearing the error packet would clean out their route cache as well.

5. Promiscuous Listening. The route cache was only updated from route reply packet, thus when a link went bad a new route discovery would have to take place. Along with the addition of multiple routes in a cache, the model was updated such that all nodes could promiscuously listen and gather route information from data packets, request packets and reply packets. This greatly improves the possibility of having a valid route available in the cache.

6. Retransmissions. In the NIST model, if an error occurred when sending a data packet the packet was automatically dropped. The specification calls for two retransmissions before packet salvaging, so this was implemented. It has been argued that this is not needed when DSR is implemented over 802.11, but experimentation showed that when the network was congested, these retransmissions were extremely beneficial.

7. Send Buffer. The send buffer is used to hold data packets waiting to be sent to their destination. The send buffer was not being checked when a route was added to the cache to see if any packets were waiting on that route information, which could unnecessarily increase the end-to-end delay of the data packets. The send buffer was also not regularly checking the packets to verify they had not expended their maximum lifetime limit. These problems were corrected for this research.

8. Random Waypoint Mobility. NIST implemented the billiard mobility model for this DSR implementation, which is described in NIST's documentation for the DSR model [PRP00]. While this is not incorrect, the billiard mobility model was not found anywhere else in the literature reviewed. The random waypoint mobility model was the model of choice for all published DSR research data. NIST had developed the random waypoint model for OPNET in its implementation of the AODV MANET routing protocol [Gue01, PRD01], so that mobility model was modified and incorporated into this DSR model.

9. Data Packet Transmission Delay Window. The DSR specification states that a source node should not send an "unbounded" number of packets along a route without the source node allowing for a route error. However, nowhere in the literature review was a transmission delay window between sending packets down the same source route specified. Thus, through experimentation as well as trial and error, an effective delay time was determined. If multiple packets are waiting in the send buffer for a particular destination, the node should wait 30 milliseconds per hop after receiving a successful acknowledgement from the next hop in the source route before sending the next data packet down that same route. This allowed enough time for the data

packet to be transmitted to the destination and for an error packet to make it back to the source node should an error occur.

10. Data Rate. The NIST DSR model implemented a 1 Mbps data rate. This normally would not have been a problem since it should be a matter of simply changing the data rate parameter to 2 Mbps to match the data rate of all other published data. However, there were implementation errors in the model when using any data rate other than 1 Mbps. These implementation errors caused the model to transmit at 5 Mbps for some packets and 1 Mbps for others even though the set data rate was 2 Mbps.

11. Jitter Delay. Jitter Delay causes a random delay between zero and 10 milliseconds for request and reply packets. This did not turn out to be of any great importance since 802.11 already implements its own random transmission delay, but it was added anyway to meet the specification call for a maximum jitter delay of 10 milliseconds.

12. RTS/CTS handshaking at the MAC layer. The RTS/CTS handshaking used by the IEEE 802.11 OPNET implementation was problematic (e.g., pointers to non-existent packets). This problem was also seen in other research using OPNET [Gue01] and had to be fixed to accurately simulate the DSR network. Without RTS/CTS an increased amount of collisions will occur causing possible transmission failures.

A.2.1 Verification and Validation Implementation. Once the DSR model had been updated to the specification standards, the verification and validation of the model was made by comparing the results to other published data [BMJ98, MBJ99, DPR01]. In particular, the results from [BMJ98] were published in [Per01], so those were used by

this research for verification and validation. The next two sections describe the parameter settings used for the model to accomplish the verification and validation. These settings were either stated explicitly in [BMJ98] or were inferred based on the research of the published data and expert opinion from pilot test simulations.

A.2.1.1 System Parameters. The system parameters were set as shown below.

1. Data Rate - A data rate of 2 Mbps was used.
2. Simulation Area - An area of 1500 x 300 meters was used for the validation and verification of this model. This area represents a highway environment with the narrow width and long length.
3. Route Cache - The size of the cache refers to the number of routes a node's cache will maintain to any particular destination node. Pilot studies showed that a cache of 50 routes to every destination produced the best results under the implemented caching strategy.
4. Node Mobility - The random waypoint model as described in [BMJ98] was implemented and used for this verification and validation of DSR.
5. Transmission Range - The nominal transmission range of the model was set to 250 meters. This is the range that was used for most of the published MANET research.

A.2.1.2 Workload Parameters.

1. Nodes - A total of 50 nodes were placed in the simulation area.

2. Source Nodes - 20 of the 50 nodes were used as data packet source generators for peer-to-peer connections.

3. Size of data packets - 64 byte packets were generated by the 20 source nodes.

4. Packet Interarrival - The data packets were generated at a constant rate of 4 packets/second.

5. Node Speed - The node speed is uniformly distributed between 0 and 20 meters/second.

6. Node Pause Time - The node pause time for the random waypoint mobility model is varied between 0, 30, 60, 120, 300, 600, and 900 seconds.

7. Hop Delay - The specification states that the Hop Delay should be twice the propagation delay and [BMJ98] mentioned that the propagation delay is 600 microseconds. Thus, the hop delay was set at 1.2 milliseconds.

8. Transmission Delay Window - If there were multiple packets waiting in the send buffer for a particular destination, the source node would wait 30 milliseconds per hop after receiving a successful acknowledgement from the next hop in the source route before sending another data packet down the same route.

Appendix B. IMMAS Goodput Ratio Allocation of Variation (ANOVA) Worksheet

Table B.1. Goodput Ratio Data

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**
20 Source Nodes	0.564996	0.619724	0.59792	0.208299	0.263917	0.22328	0.096352	0.10917	0.109591
	0.592982	0.635972	0.619145	0.232911	0.25539	0.274558	0.101538	0.114942	0.112743
	0.597221	0.614249	0.614908	0.258129	0.265349	0.249073	0.11476	0.107912	0.107786
	0.630012	0.568694	0.603443	0.254671	0.232916	0.227188	0.111407	0.103112	0.123813
	0.65085	0.594002	0.609431	0.270479	0.27123	0.222493	0.12026	0.123464	0.119216
30 Source Nodes	0.567856	0.585663	0.582765	0.23966	0.223299	0.244697	0.099285	0.097863	0.09791
	0.636311	0.615076	0.582765	0.272878	0.231976	0.251294	0.121614	0.105172	0.111542
	0.64421	0.595511	0.585242	0.25674	0.261974	0.248049	0.113781	0.116966	0.114565
	0.598655	0.61256	0.587772	0.255877	0.251933	0.255778	0.106439	0.112671	0.114227
	0.580821	0.608492	0.58291	0.242993	0.244303	0.247065	0.099642	0.119202	0.09723

Table B.2. Goodput Ratio Means of Data

	DSR Baseline			IMMAS with ECC			IMMAS with RSA				
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**	**Row Sum**	**Row Mean**
20 Source Nodes	0.6072122	0.6065282	0.6089694	0.2448978	0.2577604	0.2393184	0.1088634	0.11172	0.1146298	2.8998996	0.322211067
30 Source Nodes	0.6055706	0.6034604	0.5842908	0.2536296	0.242697	0.2493766	0.1081522	0.1103748	0.1070948	2.8646468	0.318294089
Column Sum	1.2127828	1.2099886	1.1932602	0.4985274	0.5004574	0.488695	0.2170156	0.2220948	0.2217246	5.7645464	
Column Mean	0.6063914	0.6049943	0.5966301	0.2492637	0.2502287	0.2443475	0.1085078	0.1110474	0.1108623		**0.320252578**
Column Effect	0.28138822	0.284741722	0.276377522	-0.070988878	-0.070023878	-0.075905078	-0.211744778	-0.209205178	-0.209390278		

Table B.3. Goodput Ratio Standard Deviations

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**
20 Source Nodes	0.033573124	0.025927831	0.008538929	0.024536368	0.014998197	0.022496959	0.009763694	0.007800703	0.006729627
30 Source Nodes	0.033618205	0.012479807	0.002212819	0.01316732	0.015416776	0.004291242	0.009577177	0.00880696	0.008776739

Table B.4. Goodput Ratio 90% Confidence Intervals

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**
20 Source Nodes	0.582513582	0.587453967	0.602687597	0.226847219	0.246726731	0.222768144	0.101680579	0.105981285	0.10967904
	0.631910818	0.625602433	0.615251203	0.262948381	0.268794069	0.255868656	0.116046221	0.117458715	0.11958056
30 Source Nodes	0.580838818	0.594279426	0.582662903	0.243942845	0.231355397	0.246219678	0.101106593	0.103895816	0.100638048
	0.630302382	0.612641374	0.585918697	0.263316355	0.254038603	0.252533522	0.115197807	0.116853784	0.113551552

Table B.5. Goodput Ratio Main Effects

Factor	**Variable Designation**	**Level 1**	**Level 2**	**Level 3**
Source Nodes	A	0.001958489	-0.001958489	N/A
Authentication System	B	0.282419356	-0.072305944	-0.210113411
Pause Time (Mobility)	C	0.001135056	0.001837556	-0.002972611

Table B.6. Goodput Ratio Second Order Interaction Effects

Authentication System (B)	Source Nodes (A) 20 Source Nodes	Source Nodes (A) 30 Source Nodes
DSR Baseline	0.002939511	-0.002939511
IMMAS with ECC	-0.002579589	0.002579589
IMMAS with RSA	-0.000359922	0.000359922

Pause Time (C)	Source Nodes (A) 20 Source Nodes	Source Nodes (A) 30 Source Nodes
0 sec	-0.003021656	0.003021656
60 sec	0.001287578	-0.001287578
300 sec	0.001734078	-0.001734078

Pause Time (C)	Authentication System (B) DSR Baseline	IMMAS with ECC	IMMAS with RSA
0 sec	0.002584411	0.000182011	-0.002766422
60 sec	0.000484811	0.000444511	-0.000929322
300 sec	-0.003069222	-0.000626522	0.003695744

Table B.7. Goodput Ratio Third Order Interaction Effects

	DSR Baseline 0	DSR Baseline 60	DSR Baseline 300	IMMAS with ECC 0	IMMAS with ECC 60	IMMAS with ECC 300	IMMAS with RSA 0	IMMAS with RSA 60	IMMAS with RSA 300
20 Source Nodes	-0.001055544	-0.004651678	0.005707222	-0.000723144	0.006865222	-0.006142078	0.001778689	-0.002213544	0.000434856
30 Source Nodes	0.001055544	0.004651678	-0.005707222	0.000723144	-0.006865222	0.006142078	-0.001778689	0.002213544	-0.000434856

Table B.8. Goodput Ratio Allocation of Variation

SSY	SS0	SSA	SSB	SSC	SSAB	SSAC	SSBC	SSABC	SST	SSE
13.12965056	9.230554222	0.000345211	3.874094626	0.000405041	0.000462736	0.000413859	0.000391329	0.001489566	3.899096341	0.021493973
		Var Due to Source Nodes	Var Due to Authentication System	Var Due to Pause Time	Var Due to Source Nodes and Authentication System	Var Due to Source Nodes and Pause Time	Var Due to Authentication and Pause Time	Var Due to All Factors		Var Due to Error
		0.01%	99.36%	0.01%	0.01%	0.01%	0.01%	0.04%		0.55%
DOF_Y	DOF_0	DOF_A	DOF_B	DOF_C	DOF_{AB}	DOF_{AC}	DOF_{BC}	DOF_{ABC}	DOF_T	DOF_E
90	1	1	2	2	2	2	4	4	89	16
		MSA	MSB	MSC	MSAB	MSAC	MSBC	MSABC		MSE
		0.000345211	1.937047313	0.000202521	0.000231368	0.000206929	9.78321E-05	0.000372392		0.001343373
		F_{compA}	F_{compB}	F_{compC}	F_{compAB}	F_{compAC}	F_{compBC}	$F_{compABC}$		
		0.256973309	1441.927793	0.150755321	0.172229286	0.154037055	0.072825726	0.277206288		
		F_{TableA}	F_{TableB}	F_{TableC}	$F_{TableAB}$	$F_{TableAC}$	$F_{TableBC}$	$F_{TableABC}$		
		3.05	2.67	2.67	2.67	2.67	2.33	2.33		
		P-valueA	P-valueB	P-valueC	P-valueAB	P-valueAC	P-valueBC	P-valueABC		
		0.619121888	8.58912E-19	0.861265491	0.84332612	0.858496344	0.989412583	0.888398317		

Appendix C. IMMAS End-To-End Delay Allocation of Variation (ANOVA) Worksheet

Table C.1. End-To-End Delay Data

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**
20 Source Nodes	0.002466	0.001817	0.001981	0.004509	0.002952	0.003785	0.007213	0.006186	0.006338
	0.002101	0.001629	0.00174	0.003801	0.002983	0.002308	0.007661	0.005594	0.005268
	0.001977	0.002123	0.001865	0.00247	0.002884	0.003658	0.005124	0.006651	0.005522
	0.001684	0.002277	0.001891	0.014454	0.003464	0.003394	0.005409	0.006047	0.004116
	0.001463	0.002072	0.002574	0.002305	0.00275	0.004304	0.004587	0.004442	0.005242
30 Source Nodes	0.003625	0.002248	0.002309	0.004043	0.005536	0.003956	0.011358	0.021906	0.017448
	0.00229	0.001965	0.002309	0.002764	0.004314	0.006456	0.005401	0.00964	0.007357
	0.00167	0.002252	0.00242	0.00342	0.003382	0.003552	0.01095	0.007755	0.006746
	0.00215	0.002446	0.002311	0.005475	0.004797	0.00287	0.014344	0.010774	0.01529
	0.002388	0.002059	0.002307	0.003277	0.005136	0.003672	0.014272	0.006614	0.01505

Table C.2. Natural Log of End-To-End Delay Data

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**
20 Source Nodes	-0.570936628	-0.478481061	-0.514298313	-1.568780731	-1.332120619	-1.49932869	-2.339747127	-2.214848979	-2.211000025
	-0.522591235	-0.452600742	-0.479415785	-1.457098873	-1.36496349	-1.292592747	-2.287322166	-2.163327626	-2.182644387
	-0.51546805	-0.487354896	-0.486282616	-1.354295819	-1.326709338	-1.390009253	-2.164912288	-2.226439199	-2.227607499
	-0.462016412	-0.564412775	-0.505103692	-1.367782763	-1.457077405	-1.48197741	-2.194565117	-2.271939503	-2.088982916
	-0.429476078	-0.520872593	-0.495229544	-1.307560818	-1.30478811	-1.502859639	-2.118099213	-2.091805663	-2.126818305
30 Source Nodes	-0.565887414	-0.53501074	-0.539971261	-1.428534027	-1.499243598	-1.407734569	-2.309760777	-2.324186738	-2.32370659
	-0.452067841	-0.486009442	-0.539971261	-1.29873047	-1.461121361	-1.381131711	-2.106903184	-2.252158174	-2.193354077
	-0.439730519	-0.518335418	-0.535729842	-1.359691379	-1.339510017	-1.394128972	-2.173479731	-2.145871985	-2.166612931
	-0.513069807	-0.490108382	-0.531416161	-1.363058419	-1.3785921	-1.363445398	-2.240183228	-2.183283211	-2.169567582
	-0.543312659	-0.496771514	-0.539722478	-1.414722643	-1.409346021	-1.398103819	-2.306171517	-2.126935746	-2.330675973

Table C.3. End-To-End Delay Means

	DSR Baseline			IMMAS with ECC			IMMAS with RSA				
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**	**Row Sum**	**Row Mean**
20 Source Nodes	-0.50009768	-0.500744413	-0.49606599	-1.411103801	-1.357131793	-1.433353548	-2.220929182	-2.193672194	-2.167410626	-12.28050923	-1.364501103
30 Source Nodes	-0.502813648	-0.505247099	-0.537362201	-1.372947388	-1.417562619	-1.388908894	-2.227299687	-2.206487171	-2.236783431	-12.39541214	-1.377268024
Column Sum	-1.002911328	-1.005991513	-1.033428191	-2.784051188	-2.774694412	-2.822262441	-4.44822887	-4.400159365	-4.404194057	-24.67592136	
Column Mean	-0.501455664	-0.502995756	-0.516714095	-1.392025594	-1.387347206	-1.411131221	-2.224114435	-2.200079682	-2.202097029		**-1.37088452**
Column Effect	0.869428856	0.867888764	0.854170425	-0.021141074	-0.016462686	-0.0402467	-0.853229914	-0.829195162	-0.831212508		

Table C.4. End-To-End Delay Standard Deviations

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	**0**	**60**	**300**	**0**	**60**	**300**	**0**	**60**	**300**
20 Source Nodes	0.055222184	0.043161189	0.014033362	0.103443909	0.059877806	0.091159787	0.090736833	0.068821304	0.058211702
30 Source Nodes	0.055403693	0.020786307	0.003781709	0.051518712	0.063732976	0.017139691	0.087433529	0.081363509	0.083215718

Table C.5. End-To-End Delay 90% Confidence Intervals

Pause Time (seconds) -->	DSR Baseline 0	DSR Baseline 60	DSR Baseline 300	IMMAS with ECC 0	IMMAS with ECC 60	IMMAS with ECC 300	IMMAS with RSA 0	IMMAS with RSA 60	IMMAS with RSA 300
20 Source Nodes	-0.540722784	-0.532496648	-0.506389863	-1.487204006	-1.401181881	-1.500416737	-2.287681218	-2.244301712	-2.210235017
	-0.459472577	-0.468992178	-0.485742117	-1.335003596	-1.313081705	-1.366290359	-2.154177146	-2.143042676	-2.124586235
30 Source Nodes	-0.543572281	-0.520538886	-0.540144277	-1.410847971	-1.464448826	-1.401517988	-2.291621594	-2.266343567	-2.298002436
	-0.462055015	-0.489955312	-0.534580125	-1.335046804	-1.370676413	-1.376299799	-2.162977781	-2.146630774	-2.175564426

Table C.6. End-To-End Delay Main Effects

Factor	Variable Designation	Level 1	Level 2	Level 3
Source Nodes	A	0.006383495	-0.006383495	N/A
Authentication System	B	0.863829348	-0.025950153	-0.837879195
Pause Time (Mobility)	C	-0.001647377	0.007410305	-0.005762928

Table C.7. End-To-End Delay Second Order Interaction Effects

Authentication System (B)	Source Nodes (A): 20 Source Nodes	Source Nodes (A): 30 Source Nodes
DSR Baseline	0.001702316	-0.001702316
IMMAS with ECC	-0.010078535	0.010078535
IMMAS with RSA	0.008376219	-0.008376219

Pause Time (C)	Source Nodes (A): 20 Source Nodes	Source Nodes (A): 30 Source Nodes
0 sec	-0.011228485	0.011228485
60 sec	0.006574587	-0.006574587
300 sec	0.004653899	-0.004653899

Pause Time (C)	Authentication System (B): DSR Baseline	Authentication System (B): IMMAS with ECC	Authentication System (B): IMMAS with RSA
0 sec	0.007246885	0.006456457	-0.01370334
60 sec	-0.00335089	0.002077162	0.001273728
300 sec	-0.003895996	-0.008533619	0.012429615

Table C.8. End-To-End Delay Third Order Interaction Effects

	DSR Baseline 0	DSR Baseline 60	DSR Baseline 300	IMMAS with ECC 0	IMMAS with ECC 60	IMMAS with ECC 300	IMMAS with RSA 0	IMMAS with RSA 60	IMMAS with RSA 300
20 Source Nodes	0.004500658	-0.012409054	0.007908396	-0.004154681	0.027335867	-0.023181185	-0.000345977	-0.014926813	0.015272789
30 Source Nodes	-0.004500658	0.012409054	-0.007908396	0.004154681	-0.027335867	0.023181185	0.000345977	0.014926813	-0.015272789

Table C.9. End-To-End Delay Allocation of Variation

SSY	SS0	SSA	SSB	SSC	SSAB	SSAC	SSBC	SSABC	SST	SSE
212.9525169	169.1391931	0.003667411	43.46748296	0.002725135	0.005239074	0.005728885	0.00541647	0.019948492	43.81332378	0.303115356
		Var Due to Source Nodes	Var Due to Authentication System	Var Due to Pause Time	Var Due to Source Nodes and Authentication System	Var Due to Source Nodes and Pause Time	Var Due to Authentication and Pause Time	Var Due to All Factors		Var Due to Error
		0.01%	99.21%	0.01%	0.01%	0.01%	0.01%	0.05%		0.69%
DOF_Y	DOF_0	DOF_A	DOF_B	DOF_C	DOF_{AB}	DOF_{AC}	DOF_{BC}	DOF_{ABC}	DOF_T	DOF_E
90	1	1	2	2	2	2	4	4	89	16
		MSA	MSB	MSC	MSAB	MSAC	MSBC	MSABC		MSE
		0.003667411	21.73374148	0.001362567	0.002619537	0.002864443	0.001354117	0.004987123		0.01894471
		F_{compA}	F_{compB}	F_{compC}	F_{compAB}	F_{compAC}	F_{compBC}	$F_{compABC}$		
		0.193584954	1147.219554	0.071923365	0.138272744	0.15120013	0.071477338	0.263246204		
		F_{TableA}	F_{TableB}	F_{TableC}	$F_{TableAB}$	$F_{TableAC}$	$F_{TableBC}$	$F_{TableABC}$		
		3.05	2.67	2.67	2.67	2.67	2.33	2.33		
		P-valueA	P-valueB	P-valueC	P-valueAB	P-valueAC	P-valueBC	P-valueABC		
		0.665834997	5.28937E-18	0.930901341	0.871890551	0.860889571	0.98977871	0.897226662		

Appendix D. IMMAS Throughput Allocation of Variation (ANOVA) Worksheet

Table D.1. Throughput Data

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	0	60	300	0	60	300	0	60	300
20 Source Nodes	1827.596622	1497.075911	1621.629867	4223.369778	3113.121067	3857.040533	8268.435911	7157.873778	7161.764978
	1723.699556	1463.769422	1560.613156	3806.845333	3357.8128	3042.859556	8070.098489	7031.990044	7173.825422
	1636.668089	1529.964978	1562.021511	3202.017422	3093.7328	3377.190756	6769.0496	7332.022044	7297.729422
	1482.350578	1848.197689	1619.656	3312.989333	3720.843378	3818.4288	7119.530667	7771.158756	6358.493867
	1342.389333	1652.807644	1572.987022	3007.800533	2979.530311	3873.1184	6443.008	6189.647644	6472.886044
30 Source Nodes	2727.411911	2567.779022	2608.8384	5300.748267	5780.1488	5178.029333	12033.77493	12178.06791	12182.55076
	2189.593067	2348.303644	2608.8384	4563.038044	5632.964444	5084.6208	9821.366044	11517.58791	10869.53244
	2096.803378	2478.5168	2565.076444	4835.135111	4694.498133	5080.425956	10309.65476	10032.03698	10303.35147
	2448.283378	2348.022222	2571.690667	4867.282667	4982.020444	4891.273778	11025.24871	10455.54062	10307.03787
	2597.129422	2340.994489	2559.864	5172.765689	5172.935111	5052.055111	11871.00444	9751.1424	12159.29458

Table D.2. Throughput Means of Data

	DSR Baseline			IMMAS with ECC			IMMAS with RSA				
Pause Time (seconds) -->	0	60	300	0	60	300	0	60	300	Row Sum	Row Mean
20 Source Nodes	1602.540836	1598.363129	1587.381511	3510.60448	3253.008071	3593.727609	7334.024533	7096.538453	6892.939947	36469.12857	4052.125397
30 Source Nodes	2411.844231	2416.723235	2582.861582	4947.793956	5252.513386	5057.280996	11012.20978	10786.87516	11164.35342	55632.45575	6181.383972
Column Sum	4014.385067	4015.086364	4170.243093	8458.398435	8505.521458	8651.008605	18346.23431	17883.41362	18057.29337	92101.58432	
Column Mean	2007.192533	2007.543182	2085.121547	4229.199218	4252.760729	4325.504302	9173.117156	8941.706809	9028.646685		**5116.754684**
Column Effect	-3109.562151	-3109.211502	-3031.633138	-887.5554667	-863.9939556	-791.2503821	4056.362471	3824.952124	3911.892		

Table D.3. Throughput Standard Deviations

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	0	60	300	0	60	300	0	60	300
20 Source Nodes	192.7554891	156.8712738	30.74682208	495.7021638	295.6268665	370.2118839	802.1914123	578.9957974	440.7652324
30 Source Nodes	266.4055616	102.1912521	24.081006	292.5748712	451.1159202	104.2221441	960.8571715	1027.334415	947.3472768

Table D.4. Throughput 90% Confidence Intervals

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
Pause Time (seconds) -->	0	60	300	0	60	300	0	60	300
20 Source Nodes	1460.737106	1482.958209	1564.762108	3145.933071	3035.525329	3321.375178	6743.879293	6670.59072	6568.68379
	1744.344566	1713.768048	1610.000914	3875.275889	3470.490813	3866.08004	7924.169773	7522.486187	7217.196103
30 Source Nodes	2215.85862	2341.544568	2565.145996	4732.556464	4920.642577	4980.60827	10305.33948	10031.09979	10467.4219
	2607.829842	2491.901902	2600.577168	5163.031448	5584.384195	5133.953721	11719.08008	11542.65054	11861.28495

Table D.5. Throughput Main Effects

Factor	Variable Designation	Level 1	Level 2	Level 3
Source Nodes	A	-1064.629288	1064.629288	N/A
Authentication System	B	-3083.46893	-847.5999348	3931.068865
Pause Time (Mobility)	C	19.74828444	-49.41777786	29.66949341

Table D.6. Throughput Second Order Interaction Effects

Authentication System (B)	Source Nodes (A)	
	20 Source Nodes	30 Source Nodes
DSR Baseline	627.4386924	-627.4386924
IMMAS with ECC	247.9212583	-247.9212583
IMMAS with RSA	-875.3599506	875.3599506

Pause Time (C)	Source Nodes (A)	
	20 Source Nodes	30 Source Nodes
0 sec	77.18293529	-77.18293529
60 sec	-20.07106761	20.07106761
300 sec	-57.11186768	57.11186768

Pause Time (C)	Authentication System (B)		
	DSR Baseline	IMMAS with ECC	IMMAS with RSA
0 sec	-45.84150511	-59.70381634	105.5453215
60 sec	23.67520589	33.02375706	-56.6989629
300 sec	22.16629922	26.68005929	-48.8463585

Table D.7. Throughput Third Order Interaction Effects

	DSR Baseline			IMMAS with ECC			IMMAS with RSA		
	0	60	300	0	60	300	0	60	300
20 Source Nodes	-44.64403756	48.08160984	-3.437572289	20.93035641	-162.9735604	142.043204	23.71368114	114.8919505	-138.6056317
30 Source Nodes	44.64403756	-48.08160984	3.437572289	-20.93035641	162.9735604	-142.043204	-23.71368114	-114.8919505	138.6056317

Table D.8. Throughput Allocation of Variation

SSY	SS0	SSA	SSB	SSC	SSAB	SSAC	SSBC	SSABC	SST	SSE
3286695771	2356306065	102009196.9	770385261.5	111371.7104	36641979.19	288654.5605	252607.9759	844655.0131	930389705.5	19855978.67
		Var Due to Source Nodes	Var Due to Authentication System	Var Due to Pause Time	Var Due to Source Nodes and Authentication System	Var Due to Source Nodes and Pause Time	Var Due to Authentication and Pause Time	Var Due to All Factors		Var Due to Error
		10.96%	82.80%	0.01%	3.94%	0.03%	0.03%	0.09%		2.13%
DOF_Y	DOF_0	DOF_A	DOF_B	DOF_C	DOF_{AB}	DOF_{AC}	DOF_{BC}	DOF_{ABC}	DOF_T	DOF_E
90	1	1	2	2	2	2	4	4	89	16
		MSA	MSB	MSC	MSAB	MSAC	MSBC	MSABC		MSE
		102009196.9	385192630.8	55685.85519	18320989.59	144327.2803	63151.99396	211163.7533		1240998.667
		F_{compA}	F_{compB}	F_{compC}	F_{compAB}	F_{compAC}	F_{compBC}	$F_{compABC}$		
		82.19927997	310.3892382	0.044871809	14.76310175	0.116299303	0.050888043	0.17015631		
		F_{TableA}	F_{TableB}	F_{TableC}	$F_{TableAB}$	$F_{TableAC}$	$F_{TableBC}$	$F_{TableABC}$		
		3.05	2.67	2.67	2.67	2.67	2.33	2.33		
		P-valueA	P-valueB	P-valueC	P-valueAB	P-valueAC	P-valueBC	P-valueABC		
		1.05493E-07	1.58874E-13	0.95623992	0.000232739	0.890954372	0.994642675	0.950483372		

[Ano02] Anonymous. Review Comments for Journal Submission of IMMAS, January 2002.

[BaB98] Brutch, Tasneem G. and Paul C. Brutch. Mutual Authentication, Confidentiality, and Key MANagement (MACKMAN) System for Mobile Computing and Wireless Communications. In *Computer Security Applications Conference*, volume 14th Annual, pages 308–317, 1998.

[BDS94] Bharghavan, Vaduvur, Alan Demers, Scott Shenker, and Lixia Zhang. MACAW: A Media Access Protocol for Wireless LAN's. In *Proceedings of the ACM SIGCOMM 1994 Conference*, pages 212–225, August 1994.

[BGH95] Bird, Gopal, Herzberg, Janson, Kutten, Molva, and Yung. The KryptoKnight Family of Light-Weight Protocols for Authentication and Key Distribution. *IEEETNWKG: IEEE/ACM Transactions on NetworkingIEEE Communications Society, IEEE Computer Society and the ACM with its Special Interest Group on Data Communication (SIGCOMM), ACM Press*, 3, 1995.

[BMJ98] Broch, Josh, David Maltz, David Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *Mobile Computing and Networking*, pages 85–97, 1998.

[Bol00] Boleng, Jeff. Efficient Network Layer Addressing for Mobile Ad Hoc Networks. Technical Report MCS–00-09, The Colorado School of Mines, 2000.

[BSS99] Blake, Ian, Gadiel Seroussi, and Nigel Smart. *Elliptic Curves in Cryptography.* Cambridge University Press, 1999.

[Cer97] Certicom. The Elliptic Curve Cryptosystem: An Introduction to Information Security. White Paper found at "http://www.certicom.com/resources/download/EccWhite1ps.zip", March 1997.

[CaM99] Corson, S. and J. Macker. Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. Request For Comments (RFC) 2501, Internet Engineering Task Force: MANET Working Group, January 1999.

[CMC99] Corson, M. Scott, Joseph P. Macker, and Gregory H. Cirincione. Interet-Based Mobile Ad Hoc Networking. *IEEE Internet Computing*, 3:63–70, July-August 1999.

[DaA99] Dierks, Tim and Christopher Allen. The TLS Protocol. Request For Comments (RFC) 2246, Internet Engineering Task Force (IETF) Transport Layer Security Working Group, January 1999.

[Dar90] Darn, Phil. MACA—A New Channel Access Method for Packet Radio. In *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, pages 134–140, September 1990.

[DPR01] Das, Samir Ranjan, Charles E. Perkins, and Elizabeth E. Royer. Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks. *IEEE Personal Communications*, 8:16–28, 2001.

[EWS98] Erwin, Mike, Paul Wolfe, and Charlie Scott. *Virtual Private Networks.* O'Reilly, 2nd Edition, 1998.

[GaK00] Gupta, P. and P.R. Kumar. The Capacity of Wireless Networks. *IEEE Transactions on Information Theory*, 46:388–404, March 2000.

[GRS99] Goldschlag, David M., Michael G. Reed, and Paul F. Syverson. Onion Routing for Anonymous and Private Internet Connections. *Communications of the ACM*, 42:39–41, February 1999.

[Gue01] Guemari, Lyes. An OPNET model implementation for Ad-hoc On demand Distance Vector Routing Protocol. Master's thesis at the Information Technology Laboratory of the National Institute of Standards and Technology, August 2001.

[HBC01] Hubaux, Jean-Pierre, Levente Buttyan, and Srdan Capkun. The Quest for Security in Mobile Ad Hoc Networks. In *The 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 146–155, 2001.

[HGB01] Hubaux, Jean-Pierre, Thomas Gross, Jean-Yves Le Boudec, and Martin Vetterli. Toward Self-Organized Mobile Ad Hoc Networks: The Terminodes Project. *IEEE Communications*, 39:118–124, January 2001.

[ICP00] Impett, Matthew, M. Scott Corson, and Vincent Park. A Receiver-Oriented Approach to Reliable Broadcast in Ad Hoc Networks. In *IEEE Wireless Communications and Networking Conference*, pages 117–122, September 2000.

[IEE99] Editors of IEEE 802.11. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, first edition.* Institute of Electrical and Electronics Engineers, Inc., New York, August 1999.

[IEE00] IEEE Standards Dept. *IEEE STD 1363-2000 - Standard Specifications for Public Key Cryptography.* Institute of Electrical and Electronics Engineers, Inc., New York, August 2000.

[IEE01] IEEE Standards Dept. *IEEE P1363a / D9 (Draft Version 9) - Standard Specifications for Public Key Cryptography: Additional Techniques.* Institute of Electrical and Electronics Engineers, Inc., New York, August 2001.

[JaC99] Jacobs, Stuart and Scott Corson. MANET Authentication Architecture. Internet draft, Internet Engineering Task Force (IETF) MANET Working Group, March 1999.

[JMH99] Johnson, David B., David A. Maltz, Yih-Chun Hu, and Jorjeta G. Jetcheva. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet draft, Internet Engineering Task Force MANET Working Group, October 1999. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-03.txt.

[JMH01a] Johnson, David, David Maltz, Yih-Chun Hu, and Jorjeta Jetcheva. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet draft, Internet Engineering Task Force MANET Working Group, March 2001. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-05.txt.

[JMH01b] Johnson, David, David Maltz, Yih-Chun Hu, and Jorjeta Jetcheva. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet draft, Internet Engineering Task Force MANET Working Group, November 2001. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-06.txt.

[KaN93] Kohl, J. and C. Neuman. The Kerberos Authentication Service (V5). Request For Comments (RFC) 1510, Internet Engineering Task Force: Kerberos Working Group, September 1993.

[Kob87] Koblitz, Neal. Elliptic Curve Cryptosystem. *Mathematics of Computation*, 48:203–209, 1987.

[KaS78] Kleinrock, L. and J. Silvester. Optimum Transmission Radii for Packet Radio Networks or Why Six is a Magic Number. In *Proceedings of the IEEE National Telecommunications Conference*, pages 4.3.1–4.3.5, December 1978.

[MBJ99] Maltz, D., J. Broch, J. Jetcheva, and D. Johnson. The Effects of On-Demand Behavior in Routing Protocols for Multi-Hop Wireless Ad Hoc Networks. *IEEE Journal on Selected Areas in Communication*, 17:17–25, August 1999.

[Men93] Menenzes, Alfred J. *Elliptic Curve Public Key Cryptosystems.* Kluwer Academic Publishers, 1993.

[Mil01] Miller, Sandra Kay. Facing the Challenge of Wireless Security. *IEEE Computer*, 34:16–18, July 2001.

[MTH92] Molva, Refik, Gene Tsudik, Els Van Herreweghen, and Stefano Zatti. KryptoKnight Authentication and Key Distribution System. In *Proceedings of the 1992 European Symposium on Research in Computer Security - ESORICS 1992*, pages 1–16, November 1992.

[MOV01] Menezes, Alfred J., Paul C. van Oorschort, and Scott A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 5 edition, August 2001.

[NaS78] Needham, R. and M. Schroeder. Using Encryption for Authentication in Large Networks of Computers. *Communications of the ACM*, 21:993–999, December 1978.

[Per01] Perkins, Charles E. *Ad Hoc Networking.* Addison-Wesley, 1 edition, 2001.

[PMK00] Prasad, Anand R., Henri Moelard, and Jan Kruys. Security Architecture for Wireless LANs: Corporate & Public Environment. In *51st IEEE Vehicular Technology Conference*, pages 283–287, May 2000.

[PRD01] Perkins, Charles E., Elizabeth M. Royer, and Samir R. Das. Ad hoc On Demand Distance Vector (AODV) Routing. Internet draft, Internet Engineering Task Force MANET Working Group, March 2001. http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-08.txt.

[PRP00] Pallot, Xavier, Nicolas Roux, and Jean-Sebastien Pegon. README File for NIST DSR Model. File and Model found at "http://w3.antd.nist.gov/wctg/DSRreadme.pdf", December 2000.

[PaS98] Patiyoot, D. and S.J. Shepherd. Authentication Protocols for Wireless ATM Networks. In *1st IEEE International Conference on ATM*, pages 87–96, 1998.

[RSA01] RSA Labs. Frequently Asked Questions about Today's Cryptography. Faq version 3, 2001.

[RSM01] Royer, E., P. Melliar-Smith, and L. Moser. An Analysis of the Optimum Node Density for Ad hoc Mobile Networks. Submitted for publication, 2001.

[SaA99] Stajano, Frank and Ross Anderson. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In *AT&T Software Symposium*, pages 1–11, 1999.

[Sch96] Schneier, Bruce. *Applied Cryptography.* John Wiley and Sons, Inc., 2 edition, 1996.

[TDG98] Thayer, R., N. Doraswamy, and R. Glenn. IP Security. Request For Comments (RFC) 2411, Internet Engineering Task Force: MANET Working Group, November 1998.

[VaA00] Venkatraman, Lakshmi and Dharma P. Agrawal. A Novel Authentication Scheme for Ad hoc Networks. In *IEEE Wireless Communications and Networking Conference*, pages 1268–1273, September 2000.

[VLAN98] LAN MAN Standards Committee of the IEEE Computer Society. *IEEE STD 802.1Q-1998 - IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.* Institute of Electrical and Electronics Engineers, Inc., New York, December 1998.

[YKR01] Ylonen, Tatu, Tero Kivinen, Timo Rinne, and Sami Lehtinen. SSH Authentication Protocol. Internet draft, Internet Engineering Task Force Transport Layer Security Working Group, November 2001. http://www.ietf.org/internet-drafts/draft-ietf-secsh-userauth-13.txt.

[YNK01] Yi, Seung, Prasad Naldurg, and Robin Kravets. Security-Aware Ad hoc Routing for Wireless Networks. In *The 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 299–303, October 2001.

[ZaH99] Zhou, Lidong and Zygmunt J. Haas. Securing Ad Hoc Networks. *IEEE Network*, 13:24–30, November-December 1999.

Vita

Jason Ballah is a First Lieutenant in the U.S. Air Force. He is a M.S. candidate in the Department of Electrical and Computer Engineering, Graduate School of Engineering and Management, Air Force Institute of Technology. He received his B.S. in Computer Science from Kansas State University in 1998. His technical interests include computer networking, information warfare, and information system security.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
03-01-2002	**Master's Thesis**	Jan 2001 – Mar 2002

4. TITLE AND SUBTITLE

INTEGRATED MANET MUTUAL AUTHENTICATION SYSTEM (IMMAS)

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Ballah, Jason, T., First Lieutenant, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 P Street, Building 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GCS/ENG/02M-01

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFIWC/IO (ACC)
Attn: Ms. Carol Hiltpold (IOTT)
102 Hall Blvd Ste 350 DSN: 945-3584
San Antonio TX 78243-7039 e-mail: Carol.Hiltpold@cmet.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The Integrated MANET Mutual Authentication System (IMMAS) provides implied mutual authentication of all routing and data traffic within a Mobile Ad Hoc Network (MANET) by combining Elliptic Curve Cryptography, a public-key cryptosystem, with the Dynamic Source Routing (DSR) Protocol. IMMAS provides security by effectively hiding network topology from adversaries while reducing the potential for, among other things, traffic analysis and data tampering, all while providing a graceful degradation for each of the authentication components. Current research in MANETs tends to focus primarily on routing issues leaving topics such as security and authentication for future research. IMMAS focuses on achieving a higher level of security with the potential for substantial increases in efficiency of processing power and bandwidth compared to alternative exterior authentication mechanisms tacked on after protocol development and standardization.

15. SUBJECT TERMS

Mobile Ad Hoc Network (MANET),
Mutual Authentication,
Elliptic Curve Cryptography,
Dynamic Source Routing (DSR),
Network Security

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Rusty O. Baldwin, Major, USAF (ENG)
U	U	U	UU	115	**19b. TELEPHONE NUMBER** *(Include area code)* (937) 255-3636, ext 4612; e-mail: Rusty.Baldwin@afit.edu